



Universidad de Valladolid



**ESCUELA DE INGENIERÍAS
INDUSTRIALES**

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

Grado en Ingeniería en Organización Industrial

**Creación de una herramienta de contabilidad
de costes mediante programación en VBA.**

Autor:

Requejo Jiménez, Diego

Tutor:

**Izquierdo Millán, Segismundo
Samuel**

**Dpto. Organización de
Empresas y Comercialización
e Investigación de Mercados**

Valladolid, Julio, 2020

Resumen

El objetivo de este trabajo es implementar una herramienta, lo más flexible y completa posible, basada en un sistema de gestión de contabilidad analítica. Todo esto bajo hojas de cálculo en Excel, mediante programación en Visual Basic for Applications (VBA). El usuario podrá añadir, entre otras cosas, las diferentes secciones que conforman la empresa, personal y gastos que formen el ejercicio contable en la empresa, hasta elegir los distintos tipos de imputación de costes, ya sea en uno de los posibles modos que incluye de manera predeterminada la aplicación o en un modo manual, donde el usuario elige con total precisión cómo distribuir los costes. En cuanto el usuario introduzca los parámetros que considere oportunos, la aplicación generará los distintos informes contables: Hoja analítica por secciones, Imputación de secciones a productos/servicios y el cuadro de márgenes y resultados.

Palabras clave

Contabilidad de Costes, Contabilidad Analítica, Excel, VBA, Imputación de costes.

Abstrac

The aim of this project is to implement a tool, as flexible and complete as possible, based on an analytical accounting management system. All this under Excel spreadsheets, through programming in Visual Basic for Applications. The user will be able to add, among other things, the different sections that make up the company, personnel and expenses that make up the accounting year in the company, as well as choosing different types of cost allocation, either in one of the possible modes included by default in the application or in a manual mode, where the user chooses with total precision how to distribute the costs. As soon as the user enters the parameters that are considered appropriate, the application will generate the different accounting reports: Analytical sheet by sections, Cost allocation of sections to products/services and the table of margins and results.

Keywords

Cost accounting, Analytical accounting, Excel, VBA, Cost allocation.

Índice del documento

Resumen	I
Palabras clave	I
Abstrac	I
Keywords	I
INTRODUCCIÓN Y OBJETIVOS	IX
Antecedentes y motivación.	XI
Objetivos y alcance.	XI
Estructura del Trabajo Fin de Grado.	XII
CAPÍTULO 1: LA CONTABILIDAD DE COSTES	1
Introducción a la contabilidad de costes.	3
Objetivos de la contabilidad de costes.	4
Planificación y control.	4
Valoración de los bienes y servicios.	5
Diferencias entre la contabilidad de costes y la contabilidad financiera.	6
Por su objetivos.	6
Obligatoriedad, destinatarios y formalismo.	6
Ámbito de actuación.	7
CAPÍTULO 2: GUÍA DEL PROGRAMADOR	9
Introducción.	11
Estructura cinta de opciones (Ribbon).	13
Estructura Visual Basic.	17
Formularios.	19
Módulos.	38
CAPÍTULO 3: MANUAL DE USUARIO	43
Enunciado del problema.	45
Página principal.	46
Datos de la empresa.	46
Secciones empresa.	47
Productos finales y en curso.	48
Existencias iniciales de materias primas.	50
Compras de materias primas.	50

Consumos de materias primas .	51
Facturación.	52
Personal.	53
Gastos.	54
Hoja analítica por secciones y el Configurador Manual.	55
Cuadro imputación por producto.	57
Gestión del almacén y hoja de existencias-diferencias.	58
Cuadro Márgenes y Resultados.	59
CAPÍTULO 4: ESTUDIO ECONÓMICO	61
Introducción.	63
Fases del proyecto.	63
Estudio económico.	65
Coste personal.	65
Coste material informático.	66
Costes indirectos.	66
Coste económico del proyecto.	67
Coste fase 1. Análisis previo.	67
Coste fase 2: Estimación de recursos.	68
Coste fase 3. Diseño de la herramienta.	68
Coste fase 4. Desarrollo de la herramienta.	69
Coste fase 5. Documentación.	69
Coste total del proyecto.	70
Precio de venta del producto.	71
CONCLUSIONES Y LÍNEAS FUTURAS DE DESARROLLO	73
Introducción.	75
Conclusiones.	75
Líneas futuras de desarrollo.	76
BIBLIOGRAFÍA	77
ANEXOS	81
Código Cinta de opciones.	83
Código Visual Basic.	86
Módulos.	86

Índice de ilustraciones

<i>Ilustración 1: Proceso de gestión. Basado en Sáez Torrecilla, Fernández Fernández, & Gutiérrez Díaz (2009, pág. 9)</i>	5
<i>Ilustración 2: Ámbito de actuación. Basado en Schneider (1949, págs. 8-9)</i>	8
<i>Ilustración 3: Diagrama general de la herramienta</i>	12
<i>Ilustración 4. Cinta de opciones.</i>	13
<i>Ilustración 5. Plantilla Custom UI Editor.</i>	14
<i>Ilustración 6. Grupos cinta de opciones</i>	14
<i>Ilustración 7. Macro menú.</i>	15
<i>Ilustración 8. Información botón</i>	16
<i>Ilustración 9. Hojas libro Excel</i>	18
<i>Ilustración 10. Módulos libro Excel</i>	18
<i>Ilustración 11. Formularios libro Excel.</i>	18
<i>Ilustración 12. Formulario Datos Empresa.</i>	19
<i>Ilustración 13. Mensaje de error</i>	22
<i>Ilustración 14. Formulario Estructura.</i>	22
<i>Ilustración 15. Propiedades Combobox Formulario estructura.</i>	23
<i>Ilustración 16. Error tipo imputación.</i>	25
<i>Ilustración 17. Formulario Registro de productos.</i>	26
<i>Ilustración 18. Formulario Registro productos en curso.</i>	28
<i>Ilustración 19. Formulario Existencias iniciales Materias primas</i>	30
<i>Ilustración 20. Formulario Compra Materias primas</i>	31
<i>Ilustración 21. Formulario Consumo Materias Primas</i>	33
<i>Ilustración 22. Formulario Facturación</i>	36
<i>Ilustración 23. Formulario Gastos.</i>	37
<i>Ilustración 24. Menú principal</i>	46
<i>Ilustración 25. Formulario Información.</i>	47
<i>Ilustración 26. Datos empresa</i>	47
<i>Ilustración 27. Datos Empresa completo</i>	48
<i>Ilustración 28. Formulario Registro Producto Terminado</i>	48
<i>Ilustración 29. Imagen Productos en curso en pantalla.</i>	49
<i>Ilustración 30. Formulario Registro Productos en curso.</i>	49
<i>Ilustración 31. Imagen Productos en curso en pantalla</i>	49
<i>Ilustración 32. Formulario Existencias iniciales materia prima</i>	50
<i>Ilustración 33. Imagen Existencias iniciales materia prima</i>	50
<i>Ilustración 34. Formulario compra materias prima</i>	51
<i>Ilustración 35. Imagen Compra materias primas en pantalla.</i>	51
<i>Ilustración 36. Formulario Consumo Materia prima</i>	52
<i>Ilustración 37. Consumo Materia prima en pantalla.</i>	52
<i>Ilustración 38. Formulario Facturación</i>	53
<i>Ilustración 39. Imagen facturación en pantalla.</i>	53
<i>Ilustración 40. Formulario Personal</i>	53
<i>Ilustración 41. Imagen Personal en pantalla.</i>	54
<i>Ilustración 42. Formulario gastos.</i>	54
<i>Ilustración 43. Imagen Gastos en pantalla.</i>	55

<i>Ilustración 44. Configurador sin asignar costes.</i>	55
<i>Ilustración 45. Configurador costes asignados.</i>	56
<i>Ilustración 46. Imagen hoja Analítica por secciones.</i>	56
<i>Ilustración 47. Mensaje de aviso.</i>	57
<i>Ilustración 48. Imagen Cuadro Imputación por productos.</i>	57
<i>Ilustración 49. Imagen Productos finales con coste unitario actualizado.</i>	58
<i>Ilustración 50. Mensaje de aviso.</i>	58
<i>Ilustración 51. Hoja existencia con ajustes calculados.</i>	58
<i>Ilustración 52. Imagen gestión del almacén (PMP)</i>	59
<i>Ilustración 53. Imagen cuadro de cuenta de Márgenes y Resultados.</i>	59
<i>Ilustración 54: Fases del proyecto</i>	63
<i>Ilustración 55:Gráfico costes totales por fases.</i>	70

Índice de tablas

<i>Tabla 1: Diferencia Contabilidad Financiera - Contabilidad costes (Seco, 1999, pág. 9)</i>	6
<i>Tabla 2: Diferencia Contabilidad Financiera - Contabilidad de Costes (Seco, 1999, pág. 11)</i>	7
<i>Tabla 3: Días laborales disponibles 2020</i>	65
<i>Tabla 4: Coste trabajador por hora.</i>	65
<i>Tabla 5: Coste materiales informáticos</i>	66
<i>Tabla 6. Costes indirectos</i>	67
<i>Tabla 7: Coste fase 1.</i>	67
<i>Tabla 8: Coste fase 2</i>	68
<i>Tabla 9: Coste fase 3</i>	68
<i>Tabla 10: Coste fase 4</i>	69
<i>Tabla 11: Coste fase 5</i>	69
<i>Tabla 12: Coste total del proyecto</i>	70

INTRODUCCIÓN Y OBJETIVOS

Antecedentes y motivación.

En la actualidad, vivimos en un mundo globalizado donde las empresas están en constante cambio y ajustes por la propia competitividad del mercado. Es por este motivo por el cual se ven en la obligación de realizar cambios, tanto en los productos como en los sistemas productivos para no perder en esta batalla en el mercado.

Para poder satisfacer la demanda del mercado, a la hora de querer competir en precios es necesario tener el mayor control posible y poder desgranar en qué y dónde se destinan los distintos gastos, así poder optimizar la toma de decisiones de forma coherente y justificada.

Para que esta toma de decisiones sea óptima, y con la complejidad que tiene actualmente cualquier empresa, la contabilidad de costes juega un papel imprescindible. Es esta la que se encarga de proporcionar una imagen de calidad y precisa de lo que está ocurriendo con los gastos en la empresa.

Objetivos y alcance.

En este TFG se ha marcado como principales objetivos los detallados a continuación:

El primero de ellos, el desarrollar una herramienta de contabilidad de costes lo más flexible posible, dando al usuario final la capacidad total de añadir todo cuanto sea necesario, sin verse limitado por funcionalidades de la aplicación. El segundo objetivo, que está completamente ligado con el anterior, es la realización de esta herramienta con el uso exclusivo de VBA, para poderle dar toda esa flexibilidad a la herramienta, mediante formularios y macros.

Para poder llegar a la consecución de estos objetivos, es necesario marcarse los siguientes objetivos secundarios:

- Comprender con claridad los conceptos de la contabilidad de costes.
- Aprender los fundamentos de la programación en VBA, así como una correcta implementación de este lenguaje, para que la herramienta funcione sin problemas.

Estructura del Trabajo Fin de Grado.

Después de una breve introducción, así como definir el alcance y objetivos del proyecto, en el capítulo uno se da una introducción teórica a la contabilidad de costes, su definición, objetivos, así como las diferencias que presenta con la contabilidad financiera.

A lo largo del capítulo dos, se ha desarrollado la “Guía del Programador”. En ella se ha explicado en profundidad la estructura de la herramienta, desde la programación de la cinta de opciones (Ribbon), los distintos formularios y las numerosas macros que hacen posible el funcionamiento de la herramienta. Todo ello perfectamente comentado tanto para facilitar la comprensión del documento como para posibles actualizaciones y mejoras.

En el capítulo tres, llamado “Manual de usuario”, mostramos el potencial y el funcionamiento al detalle de la herramienta, resolviendo un ejercicio completo, paso a paso, ilustrando con capturas todas las pantallas que nos va mostrando el programa, sumado a los distintos comentarios y explicaciones que hagan falta para completar la explicación.

Para finalizar el desarrollo del TFG, en el capítulo cuatro hemos realizado un estudio económico del proyecto. En este capítulo hemos calculado el coste de realizar esta herramienta como si fuera un trabajo para un cliente, obteniendo el coste unitario por producto, así como el precio de venta correspondiente para garantizar un margen de beneficio óptimo.

Una vez finalizado el desarrollo del trabajo del fin de grado, se han dedicado un punto a presentar las conclusiones y futuras líneas de trabajo.

Para concluir, se añade un apartado de Bibliografía, donde se incluye todas las fuentes de información utilizadas para el correcto desarrollo de este trabajo. Como paso previo al desarrollo del TFG se ha desarrollado una búsqueda bibliográfica para localizar los trabajos más relevantes existentes en las Bases de Datos (Nacionales e Internacionales) a las cuales está suscrita la UVa. Como consecuencia de esta búsqueda se han localizado varias de las obras referenciadas en la bibliografía, así como algunos TFG previos relacionados que han servido de referencia (González, 2015) (Castro, 2018) (Fraile, 2018).

CAPÍTULO 1: LA CONTABILIDAD DE COSTES

Introducción a la contabilidad de costes.

La contabilidad de costes surge ante la ineficiencia de transmitir información para la toma de decisiones dentro de la empresa. Es por ello por lo que la contabilidad se tuvo que reinventar en su campo de actuación, con el objetivo de poder recoger y suministrar toda la información precisa para permitir un control de costes.

Como resultado de esta ampliación surgió una nueva variante de contabilidad. Esta contabilidad tenía como objetivo primario el cálculo de las diferentes actividades que se realizan a nivel interno en la empresa, así como el coste de los productos/servicios, con la finalidad de conocer la rentabilidad de los mismos. (Fullana Belda & Ortega Paredes, 2008)

Para poder entender con claridad dónde actúa la contabilidad analítica de costes, es indispensable mencionar el estudio de los procesos internos y la cadena de valor. Es por ello que las diferentes actividades que conforman la empresa deben ser vistas como la suma de los diferentes procesos con el fin de obtener valor.

Toda empresa, ya sea de prestación de servicios o de producción y venta de productos, dispone una serie de recursos que forman la infraestructura empresarial. Todo este entramado de recursos va desde la maquinaria para la producción, como el material de oficina, hasta los diferentes trabajadores que se encuentran en ella, sin dejar de mencionar las distintos materiales y materias para la creación de estos productos o servicios.

El conjunto de todos y cada uno de los recursos que forman la empresa, se ven involucrados en la consecución del mismo fin, la creación de valor.

Esta imagen de la empresa ha sido comentada por Carmen Fullana Belda y José Luis Paredes Ortega (Fullana Belda & Ortega Paredes, 2008, pág. 24): "La empresa así entendida como un proceso de transformación de valores sugiere la idea de una cadena que va incorporando valor en cada eslabón mediante la transformación de los recursos iniciales en otros hasta obtener, al final de la cadena, el producto o servicio".

Objetivos de la contabilidad de costes.

La contabilidad de costes se mantiene en una continua evolución en el tiempo. En todos estos avances se podría indicar los siguientes objetivos (Sáez Torrecilla, Fernández Fernández, & Gutiérrez Díaz, 2009, págs. 8-9):

Planificación y control.

La contabilidad de Costes se ve envuelta en el proceso de gestión que realizan las diferentes empresas, es por ello por lo que pasamos a explicar en qué consiste este proceso de gestión.

Lo primero de todo es mencionar la planificación estratégica de la empresa. Esta es la parte del proceso de gestión en el cual se concretan los objetivos a largo plazo y cuáles son las acciones para conseguirlas.

En cuanto al control de gestión, es el proceso más largo en el proceso de gestión, el cual guía a los responsables de cumplir con la planificación estratégica para conseguir que la gestión cumpla con el plan estratégico. El control de gestión aparece de una manera clara y concisa en el presupuesto.

El siguiente punto que se encuentra dentro del proceso de gestión es el control de las tareas. Esta parte del proceso ejerce un control sobre la puesta en práctica de las decisiones tomadas en lo que al plan de gestión hablado con anterioridad. Para realizar correctamente un control de las tareas, se nutre de la información sobre lo que ocurre en la empresa, controlando, valga la redundancia que se cumple con lo estipulado en ellos presupuestos. Es posible que en ocasiones no se cumpla en su totalidad, ocasionando desviaciones, las cuales tendrán que interpretarse correctamente y generar nuevas acciones correctoras para seguir con el plan establecido.

En la siguiente imagen podemos ver una pequeña esquematización de lo que conforma un proceso de gestión:

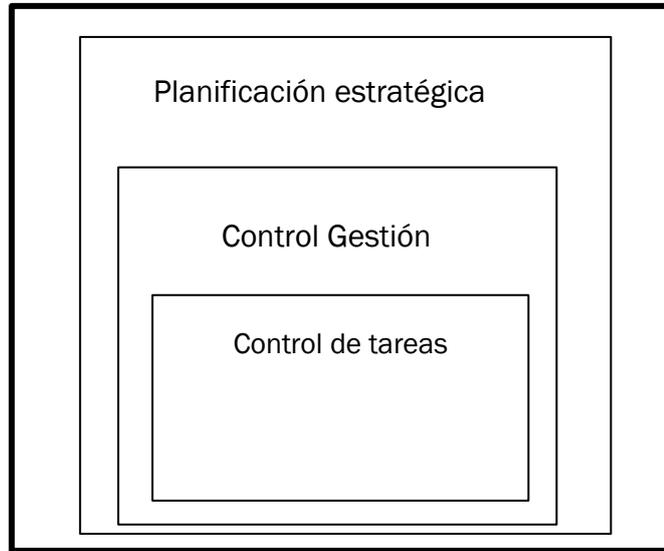


Ilustración 1: Proceso de gestión. Basado en Sáez Torrecilla, Fernández Fernández, & Gutiérrez Díaz (2009, pág. 9)

Como podemos ver en la Ilustración 1, los distintos niveles que encontramos en el proceso de gestión y lo más importante, cómo cada uno de ellos entra dentro del anterior. Es ahí donde la Contabilidad de Costes gana importancia, suministrando la información necesaria para que este proceso de gestión se lleve a cabo con éxito.

Valoración de los bienes y servicios.

Otro de los objetivos que se encuentran dentro de la Contabilidad de Costes, es la valoración de los bienes y servicios que se obtienen a lo largo del proceso productivo. Al conseguir este objetivo, logra tener la valoración de los activos que quedan al final del periodo en la empresa, como, por ejemplo:

-Materias primas. Conociendo las existencias finales de las mismas, se puede deducir con facilidad su precio de factura, aunque hay que tener en cuenta y determinar, las posibles desviaciones por las mermas sufridas en las mismas, así como el costo de transporte, adquisición...

-Productos en curso. Estos productos, en el momento que se cierra el ejercicio económico continúan en la cadena de producción.

-Productos semiterminados, productos que se encuentran entre algún centro de actividad, esperando a pasar al siguiente.

-Productos terminados. Productos los cuales han terminado todo el proceso productivo, pero aún no han sido vendidos.

Si nos referimos a una empresa de servicios, la cual no produce productos materiales, es posible pensar que entonces la Contabilidad de Costes no tiene lugar. Sin embargo, es necesario calcular su coste igualmente que en un producto material, y este coste es determinado por la Contabilidad de costes.

Diferencias entre la contabilidad de costes y la contabilidad financiera.

Es posible que cuando se habla de contabilidad, se piense automáticamente en la contabilidad financiera, mucho más conocida por la sociedad que la contabilidad de costes. Es por ello por lo que en este apartado se busca identificar las diferencias entre ambas contabilidades.

Por su objetivos.

Mientras que la contabilidad financiera centra sus esfuerzos en el cálculo del beneficio y el valor del patrimonio de la empresa, le contabilidad analítica se focaliza en la eficacia y el cálculo de los costes, así como el rendimiento en los diferentes productos de la empresa. En este punto cabe mencionar la definición que dio Julio Ortega Seco (Seco, 1999, pág. 9): “...mientras una tiene una perspectiva de la empresa eminentemente financiera y patrimonial, la otra contempla a la empresa desde una óptica económica [...], proporciona a la dirección informaciones objetivas sobre la eficiencia de las distintas alternativas.”

Contabilidad financiera	Contabilidad de costes
Cálculo beneficio Valor patrimonio	Calcula eficiencia Calcula coste Calcula rendimientos

Tabla 1: Diferencia Contabilidad Financiera - Contabilidad costes (Seco, 1999, pág. 9)

Obligatoriedad, destinatarios y formalismo.

En este punto encontramos una notoria diferencia entre ambas contabilidades. Es cierto que toda empresa se encuentra con la obligatoriedad de tener contabilidad. La diferencia es que, mientras la Contabilidad Financiera debe ser obligatoria, la Contabilidad de costes es totalmente voluntaria por la empresa.

Sobre la obligatoriedad de la contabilidad financiera ya se ha hablado en otras publicaciones, como por ejemplo hizo Julio A. Ortega Seco:

...viene motivada por los destinatarios de la información que genera cada una de las ramas contables: Mientras que la de Gestión se dirige exclusivamente a los encargados de la adopción y desarrollo de las decisiones dentro de la empresa, y nunca sobrepasa las paredes de la misma, la financiera es prácticamente pública [...] y son solicitadas y conocidas por acreedores, proveedores, clientes, sindicatos, Administración, consumidores y público en general. No debemos olvidar que la Contabilidad Financiera tiende hacia terceros, mientras que la Contabilidad de Costes es denominada Interna por la simple razón de que las conclusiones y resultados obtenidos en su seno rara vez trascienden los límites de la empresa. (Seco, 1999, pág. 10)

Es por esto por lo que la propia contabilidad Financiera es mucho más rígida, sujeta bajo una normativa y propiamente reglada, por contraposición la de Costes es totalmente flexible, con todo lo que ello conlleva.

Contabilidad Financiera	Contabilidad Costes
Obligatoria	Voluntaria
Reglada	Libre
Destinatarios externos	Flexible
Formal	Destinatarios Internos
	Informal

Tabla 2: Diferencia Contabilidad Financiera - Contabilidad de Costes (Seco, 1999, pág. 11)

Un aspecto importante a reseñar es el aspecto tributario: La empresa debe enviar al fisco su Contabilidad Financiera, y ésta debe coincidir con la fiscal, no así con la contabilidad de Costes. Esta aclaración puede ser redundante, ya que se ha indicado con anterioridad que la Contabilidad de Costes es meramente voluntaria e interna, pero se ha considerado importante aclararlo.

Ámbito de actuación.

En lo referente a su ámbito de actuación, ya se ha mencionado en el punto anterior las notables diferencias que se presentan entre ambas contabilidades. Si bien se podría resumir en que la contabilidad de financiera su ámbito es externo, ya que sus destinatarios son empresas con terceros, frente a la contabilidad de Costes que su ámbito es el interno, pues la información que transmite solo es usada en la toma de decisiones.

Para comprender mejor esto, recogemos un gráfico muy acertado junto con el comentario del mismo hecho por el autor Erich Schneider (Schneider, 1949):

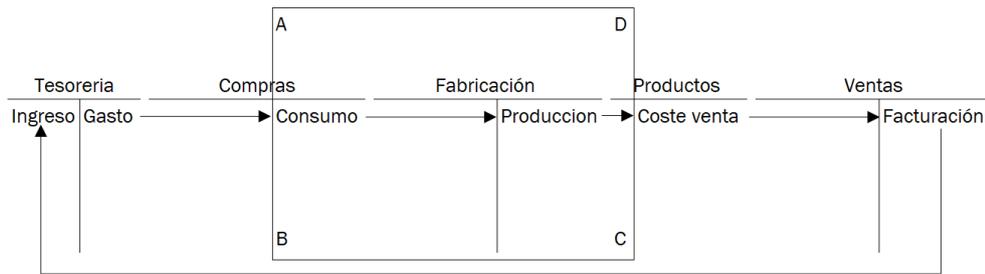


Ilustración 2: Ámbito de actuación. Basado en Schneider (1949, págs. 8-9)

Todos los asientos que se hacen fuera de la zona encerrada por la línea ABCD [...] se refieren únicamente a transacciones con el mundo exterior; en cambio, todos los asientos que caen dentro del terreno acotado se refieren solo a las transacciones que tienen lugar dentro de la producción. No son, como los primeros, consecuencia de transacciones de compra y venta, sino que son de realización de la producción. Aquella parte de la circulación que se encuentra fuera de ABCD, o sea, la referente a las transacciones de la empresa con el mundo exterior, la denominamos “parte exterior” de la circulación. En el centro de la parte exterior se encuentra los “gastos” y los “ingresos, y además sus equivalentes: “compras” y “ventas”. El tratamiento contable de la parte exterior de la circulación se hace en la contabilidad externa, cuyo centro está constituido por la contabilidad comercial. Aquella parte de la circulación que cae dentro de la zona ABCD, o sea, la que se refiere al proceso de producción, la llamamos “parte interna” de la circulación. En el punto central de la parte interna están los costes y la producción. El tratamiento contable de esta parte lo realiza la contabilidad interna. (Schneider, 1949, págs. 8-9)

CAPÍTULO 2: GUÍA DEL PROGRAMADOR

Introducción.

Comenzamos la guía del programador explicando la estructura seguida en el desarrollo de la aplicación. En este punto tenemos que aclarar que en el presente proyecto encontramos dos estructuras diferentes, la primera corresponde con el desarrollo de toda la aplicación, basada en VBA y creada con el mismo complemento que incorpora Excel en el paquete “desarrollador” (Amelot, 2016).

La segunda estructura hace referencia a la cinta de opciones de Excel, que con el objetivo de dar mayor comodidad al usuario y crear una herramienta lo más profesional posible, se ha optado por crear una nueva pestaña en la cinta de opciones donde incluir todos los botones que dispone esta herramienta. El código de la cinta de opciones está basado en XML, y ha sido necesario utilizar el programa “Custom UI Editor For Microsoft Office” para escribir dicho código.

Para facilitar la comprensión de la herramienta, se ha creado un diagrama lo más sintetizado posible con el funcionamiento de la misma, para que el lector pueda ver cuál es la estructura general de la aplicación.

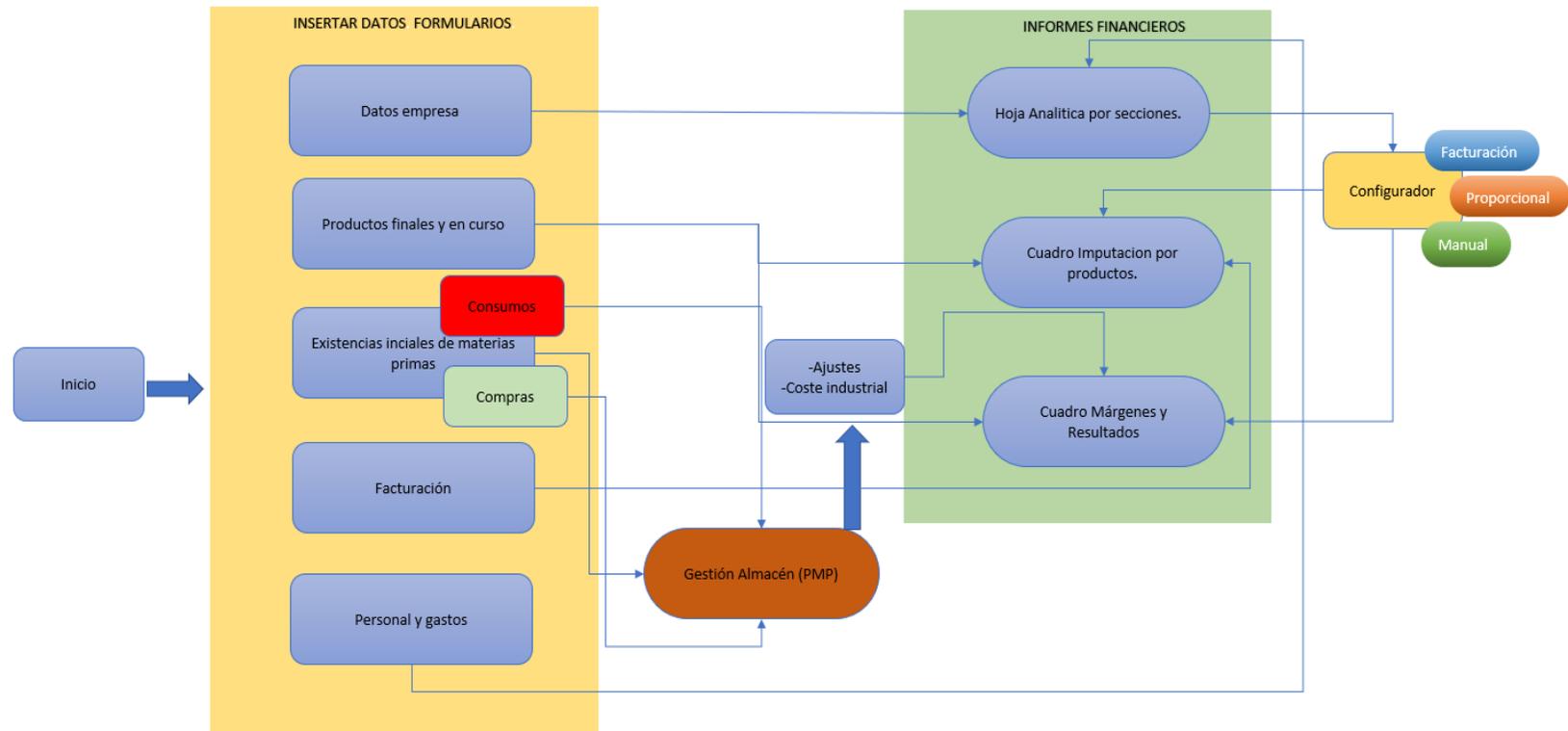


Ilustración 3: Diagrama general de la herramienta

Como podemos ver en la ilustración anterior, la herramienta se apoya en varios formularios para recoger los distintos datos. Una vez tenemos todos los datos introducidos en la aplicación, genera tanto la gestión del almacén, los ajustes y el coste industrial “real”, así como los diferentes informes financieros. Estos controlados por el configurador, el cual permite elegir los tres tipos de imputación.

Estructura cinta de opciones (Ribbon).

Como ya se ha indicado en el punto previo, en la herramienta desarrollada se ha considerado oportuno implementar una nueva pestaña, dentro de Excel, que recoja todas las funcionalidades de la aplicación. El objetivo de haber implementado esta nueva cinta de opciones no solo tiene un sentido estético, sino dotar a la herramienta de una interfaz más funcional e intuitiva, facilitando el uso de la misma (Programacion office VBA, 2019).

En la siguiente imagen se puede ver como quedaría la nueva cinta de opciones, con todos y cada uno de los botones.

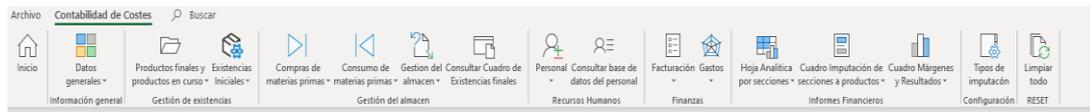


Ilustración 4. Cinta de opciones.

Si pinchamos en alguno de los botones, se ejecuta, o bien, un formulario o macro con las distintas opciones, o bien, un botón que nos mueve a la hoja correspondiente, pero eso lo desarrollaremos más adelante en la Guía del usuario.

Para poder modificar la cinta de opciones he tenido que utilizar un nuevo programa, *Custom UI Editor For Microsoft Office*, el cual cargaba el código de la cinta de opciones y permitía modificarla a nuestro antojo.

El código que utiliza es XML, el cual es muy simple y sencillo, el único problema que hemos encontrado en el desarrollo de la cinta de opciones es que el programa no tiene compilador, lo cual no indica si hay algún error en la sintaxis ni dónde está el mismo. En el caso de que hubiera un error, simplemente no se cargaría la cinta de opciones en Excel, teniendo que buscar manualmente el error, totalmente a ciegas.

Me gustaría explicar, con el objetivo de facilitar al lector del documento la comprensión del código XML de mi proyecto.

Para comenzar a modificar la cinta de opciones nos ayudamos de una plantilla de las varias que ofrece *Custom UI Editor For Microsoft Office*. En concreto partimos de la siguiente.

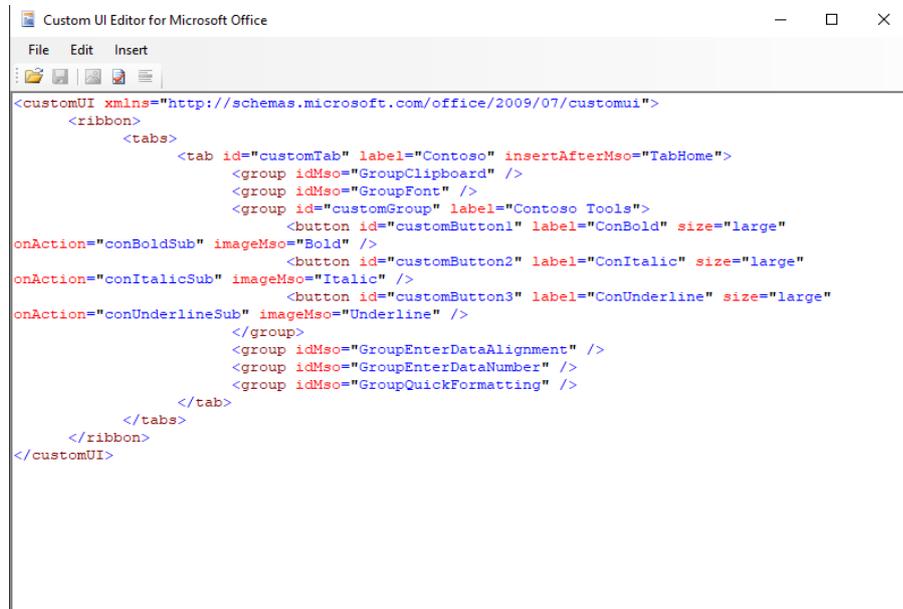


Ilustración 5. Plantilla Custom UI Editor.

Como se puede apreciar, la estructura es bastante simple. Comenzaríamos editando la línea:

```
<tab id="customTab" label="Contoso" insertAfterMso="TabHome">
```

En esta línea editamos la pestaña principal del Excel, la que saldrá en la parte superior de nuestro programa. En nuestro programa, simplemente le hemos cambiado el nombre a la variable "label" por "Contabilidad de Costes".

La siguiente estructura a explicar son los grupos, dentro de estos podremos añadir tantos botones como queramos. Mostramos en la siguiente ilustración un recorte de nuestra cinta de opciones con tres grupos:



Ilustración 6. Grupos cinta de opciones

Aquí podemos diferenciar tres grupos (Información general, Gestión de existencias, Gestión del almacén), siendo fácil localizarlos ya que el propio Excel lo separa con una línea vertical.

Como podemos ver, en el primer grupo solo tenemos un botón, en el segundo dos y, por último, en el grupo de gestión del almacén tenemos cuatro botones.

La estructura básica de cualquier grupo es la siguiente.

```
<group id="customGroup" label="Contoso Tools">
    <button id="customButton1"
    label="ConBold" size="large" onAction="conBoldSub" imageMso="Bold"
    />
</group>
```

En este caso tendríamos un grupo “Contoso Tools”, con un botón “ConBold”. Si quisiéramos añadir más botones, simplemente deberíamos añadir cuantos quisiéramos dentro del grupo.

En cuanto a lo referente a los grupos no tendría más complicación, donde encontramos más trabajo es el la creación de los diferentes botones.

En ellos debemos incluir tanto las imágenes que queremos que tenga el botón, tamaño del botón, y asignarle la acción que queremos encomendarle a dicho botón.

La estructura básica de un botón es la siguiente:

```
<button id="customButton1" label="ConBold" size="large"
onAction="conBoldSub" imageMso="Bold" />
```

Label, es el nombre que tendrá el botón, size es el tamaño del botón, y onAction, donde debemos incluir la macro que queremos que dicho botón ejecute, como, por ejemplo:

```
<group id="customGroup">
    <button id="customButton"
    label="Inicio" imageMso="BlogHomePage" size="large"
    screentip="Menú principal" onAction="menu"/>
</group>
```

En este caso tendríamos el primer grupo, el cual no tiene nombre. Dentro de este grupo tenemos un botón con el nombre “Inicio”, y como podemos ver, como onAction=” menu”.

Si nos vamos al desarrollador de Excel, vemos que la macro llamada “menú” es la siguiente:

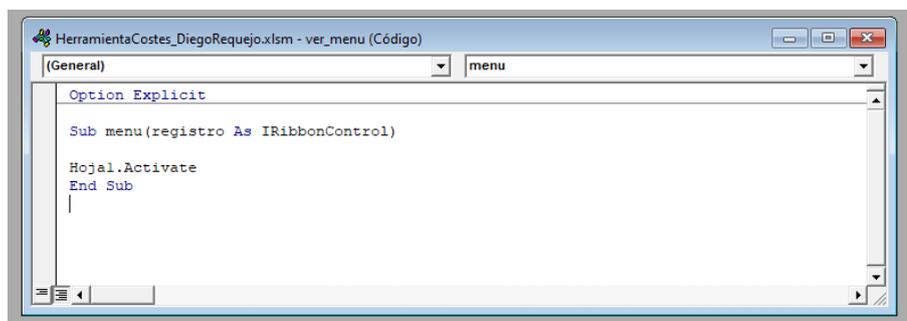


Ilustración 7. Macro menú.

Como podemos ver en la ilustración 7, tenemos la macro Sub menu, la cual coincide con el “onAction” del editor XML. Al pulsar el botón Inicio, se ejecutará esta macro, la cual activa la hoja1 de nuestro Excel.

Esta es la mecánica seguida en todos los botones. Hay que comentar que, para poder controlar las macros desde la cinta de opciones, se ha debido añadir, como aparece en la ilustración 7, el argumento a la función (registro As RibbonControl),

Continuando, explicando las distintas opciones de los botones, podemos ver como aparece screentip="Menú principal", esto lo que nos ofrece que al pasar el ratón por encima del botón “Inicio”, nos mostrará este título. Si, además, como ocurre en otros botones de la cinta de opciones, se añade screentip="textoinformativo”, nos aparecerá un recuadro con el texto informativo que queramos.

Veamos un ejemplo a continuación:

```
<button id="customButton2" label="Estructura"
imageMso="ProjectManageDeliverables" screentip="Estructura de la
empresa"
supertip="Añadimos las
distintas secciones, así como se asignan los costes en ellas."
onAction="lanzar_fm_estructura"/>
```

Este botón se encuentra dentro del grupo de “Datos Generales”, en concreto es el botón de “Estructura”. Si vamos a nuestra aplicación Excel, vemos que nos aparece lo siguiente.

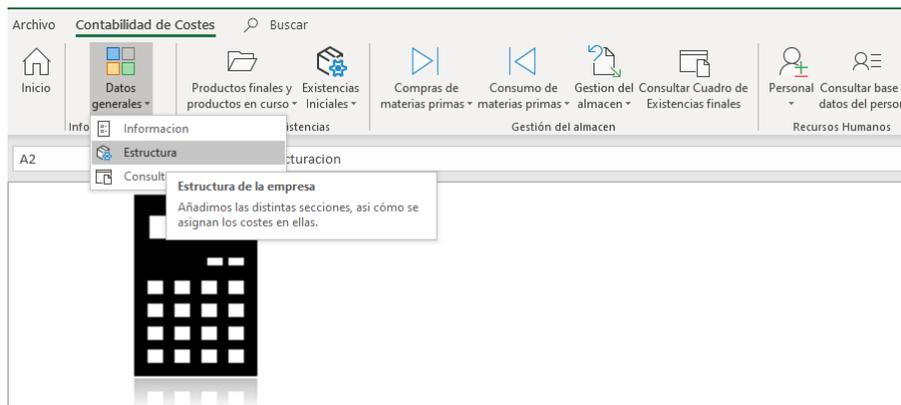


Ilustración 8. Información botón

Para finalizar de explicar todo lo referente a los botones, quisiera hacer mención a la opción imageMso, es donde añadimos las imágenes de los distintos botones. Todas las imágenes que se han añadido a los distintos botones han sido sacadas de (Bert toolkit, 2018)

Por último, y para terminar de explicar todas las estructuras utilizadas en este proyecto nos quedaría hablar de los menús. Por explicar el concepto de una manera simple, en un menu podemos agregar varios botones, dentro de un botón. Como ejemplo tenemos la ilustración 8, donde vemos que tenemos un botón principal “Datos generales” y al pulsar en él, se despliega un menu con tres botones más.

```

<group id="customGroup1" label="Información
general">
    <menu id="MyMenu2" label="Datos
generales" imageMso="CategorizeMenu" screentip="Datos generales
descripción general de empresa" supertip="Podemos añadir
la información general
description label empresa, desde la dirección, como la estructura de
la misma" size="large">
        <button id="customButton1"
label="Informacion " imageMso="PropertySheet" screentip="Datos
generales" supertip="Nombre de la entidad, dirección."
onAction="lanzar_fm_datosempresa"/>
        <button id="customButton2"
label="Estructura" imageMso="ProjectManageDeliverables"
screentip="Estructura de la empresa"
supertip="Añadimos las
distintas secciones, así como se asignan los costes en ellas."
onAction="lanzar_fm_estructura"/>
        <button id="customButton211"
label="Consultar Datos Generales" imageMso="DatabaseDocumenter"
screentip="Acceder base de datos" onAction="verempresa"/>
    </menu>
</group>

```

En el código anterior tendríamos todo el grupo de información general, dentro el menu “Datos generales” y por último los tres botones que forman el desplegable, igual que podemos ver en la ilustración 8.

La lógica seguida en el resto del código para modificar la cinta de opciones ha seguido la misma estructura. Es interesante comentar que tanto para cualquier grupo, menú y botón, el comando “id” debe ser único, ya que es el identificador individual. Si esto no ocurriera, daría problemas el programa y no se ejecutaría.

El código completo de la cinta de opciones lo encontramos en los anexos como Código Cinta de opciones.

Estructura Visual Basic.

Comencemos hablando de la estructura de la aplicación. En esta debemos diferenciar tres tipos de estructuras que conforman la herramienta: Formularios, módulos y hojas, los cuales se presentan en las siguientes imágenes.

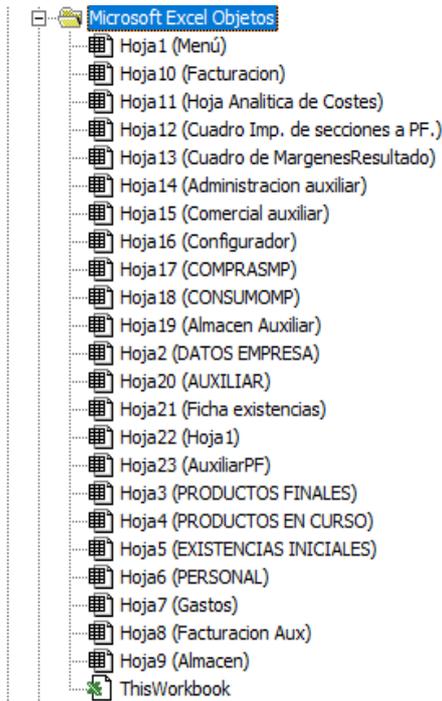


Ilustración 9. Hojas libro Excel



Ilustración 10. Módulos libro Excel

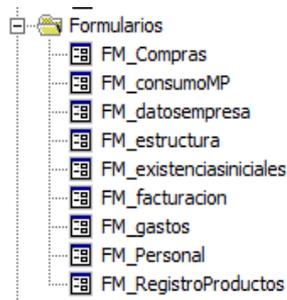


Ilustración 11. Formularios libro Excel.

Como se puede ver en la Ilustración 9, la aplicación consta de 23 hojas de Excel, que van desde la hoja 1, siendo esta el Menú, hasta las diferentes hojas para anotar los datos y realizar los distintos cálculos. Se han utilizado seis hojas auxiliares para facilitar la estructura del código.

En la ilustración 10 encontramos todas las macros que han sido necesarias para la realización del proyecto. Se ha seguido la siguiente estructura; las que comienzan por "Ax_" (siendo "X" un número) son las macros más importantes en cuanto a programación, ya que ellas realizan todos los cálculos, desde la gestión del almacén hasta la cuenta de márgenes y resultados, comenzando a nombrarlas con "Ax_" para que aparezcan arriba de la sección Módulos, ya que son las más usadas. A continuación vemos varios módulos que comienzan con "Lanzar_", todos estos módulos han sido nombrados así por que su función es

de botón para lanzar los distintos formularios desde la cinta de opciones, esto lo explicaremos más en profundidad en el siguiente punto.

Por último, vemos los módulos que comienzan por “ver_”, que tienen mucha similitud con la anterior, actuando como botón en la cinta de opciones, llevandonos a la hoja correspondiente.

Para finalizar, vemos en la ilustración 10 , los nueve formularios que necesita la herramienta para recoger los distintos datos de la empresa.

Formularios.

En este apartado entraremos a profundizar en los formularios que se han implementado en la aplicación. Si bien es cierto que la programación necesaria, en términos generales, se repite en prácticamente cada uno de ellos, ya que el concepto siempre es el mismo; recoger dato, apuntarle en la hoja correspondiente y, en algún caso, realizar alguna operación (Wiki Openoffice, 2013).

Formulario FM_DatosEmpresa.

Comenzamos hablando del formulario que recoge los parámetros más generales de la empresa.

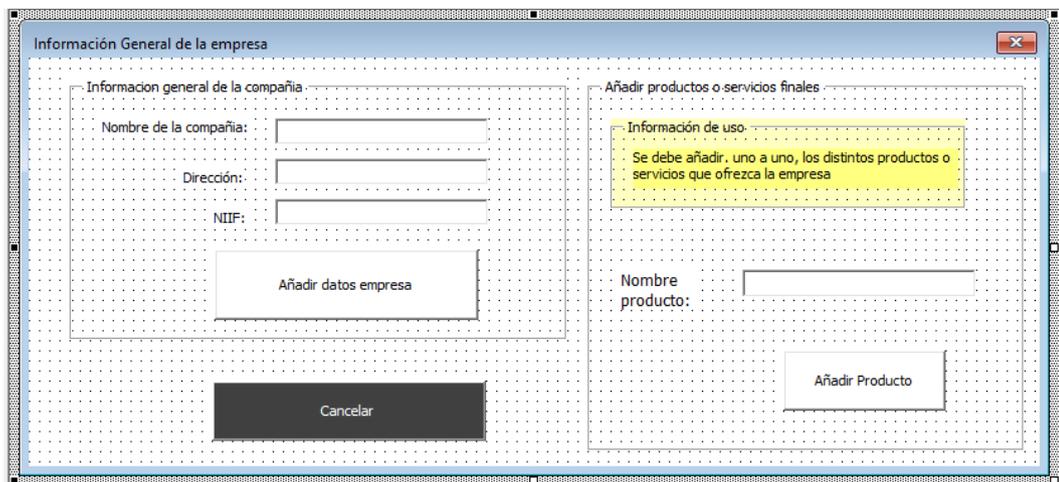


Ilustración 12. Formulario Datos Empresa.

En este formulario encontramos dos zonas diferenciadas, con dos botones para añadir distintos datos. En uno añadiremos los datos generales de la empresa

(Nombre, Dirección, NIF) y en la zona de la derecha añadimos, uno a uno los productos o servicios que ofrezca la compañía.

En cuanto al código VBA que se encuentra cuando damos clic a “Añadir datos empresa”:

```
Private Sub CommandButton1_Click()  
    Dim fila As Integer  
    Dim final As Integer 'Detectar el último registro  
  
    Hoja2.Activate  
  
    For fila = 1 To 1000  
        If Hoja2.Cells(fila, 1) = "" Then  
            final = fila  
            Exit For  
        End If  
    Next  
  
    Hoja2.Cells(final, 1) = Me.text_nomempresa  
    Hoja2.Cells(final, 2) = Me.text_dirempresa  
    Hoja2.Cells(final, 3) = Val(Me.text_nifempresa)  
  
    'borrar los datos introducidos para poder ingresar nuevos sin  
    tener que borrar  
    Me.text_nomempresa = ""  
    Me.text_dirempresa = ""  
    Me.text_nifempresa = ""  
  
End Sub
```

Esta será la estructura general de todos los botones para añadir datos. Se activa la página donde se quiere guardar la información, se realiza un bucle for para calcular cuál es la última fila, y por último se van pegando los valores según corresponda.

Si vemos el código VBA bajo el botón añadir producto, encontramos lo siguiente:

```
Private Sub CommandButton3_Click()  
    Dim fila As Integer  
    Dim final As Integer 'Detectar el ultimo registro  
    Dim registro As Integer  
  
    Hoja2.Activate  
  
    For fila = 1 To 1000  
        If Hoja2.Cells(fila, 7) = "" Then
```

```

        final = fila
    Exit For
End If
Next

For registro = 2 To final 'solo recorre lo que tenemos, no
hasta 1000, bucle para ver si hay algun registro repetido
    If Hoja2.Cells(registro, 7) = Me.text_nombreproducto Then
        MsgBox "Registro ya existente"
        Exit Sub

        Exit For
    End If
Next

ActiveSheet.Cells(final, 7) = Me.text_nombreproducto

'borrar los datos introducidos para poder ingresar nuevos sin
tener que borrar

Me.text_nombreproducto = ""
End Sub

```

Como se puede apreciar, el código es muy similar, pero se ha añadido una nueva funcionalidad, que analizamos con más detalle:

```

For registro = 2 To final 'solo recorre lo que tenemos, no hasta 1000,
bucle para ver si hay algun registro repetido
    If Hoja2.Cells(registro, 7) = Me.text_nombreproducto Then
        MsgBox "Registro ya existente"
        Exit Sub

        Exit For
    End If
Next

```

Dentro del bucle for, pese a que la intención y la estructura es la misma, se ha añadido una condición:

```

If Hoja2.Cells(registro, 7) = Me.text_nombreproducto Then
    MsgBox "Registro ya existente"
    Exit Sub

    Exit For
End If

```

Con este pequeño bucle, controlamos que no se incluye dos veces el mismo producto o servicio. Este bucle de control se ha implementado en todos los formularios para evitar errores. En caso de que se produzca un error, el programa nos sacará el siguiente mensaje por pantalla (Celemín, 2019):



Ilustración 13. Mensaje de error

Formulario FM_Estructura.

Con este formulario queremos recoger las diferentes secciones que forman parte del proceso productivo de la empresa, así como el tipo de imputación de las mismas.

En cuanto a los tipo de imputación presentamos tres opciones:

- Por facturación: Se repartirá el coste de la sección según lo facturado en cada producto.
- Proporcional: El coste de la sección se reparte proporcionalmente entre todos los productos o servicios.
- Manual: Esta es la opción más interesante, ya que permite un control total del reparto por producto. En una nueva hoja, llamada Configurador, permite asignar el porcentaje que se asigna a cada producto según el usuario considere oportuno.

La opción de tipo de imputación proporcional solo está disponible para los costes no operativos, como son Administración y Comercial.

Una vez explicado esto, mostramos el mencionado formulario en la siguiente ilustración:

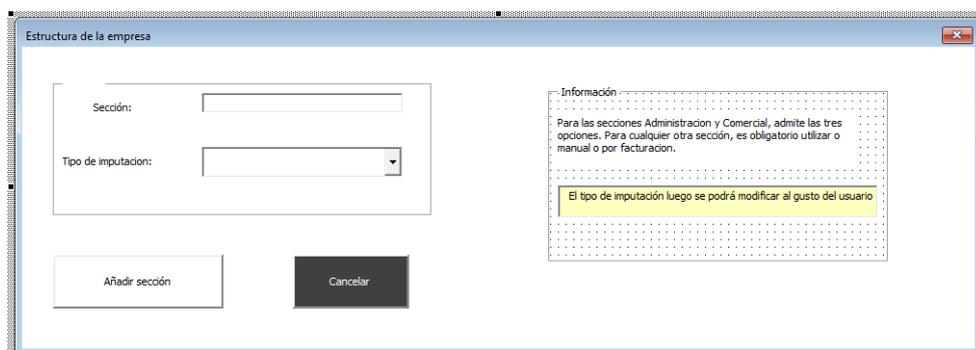


Ilustración 14. Formulario Estructura.

En este caso tenemos un formulario muy sencillo, donde podemos añadir la sección y un desplegable con los distintos tipos de imputación para mayor comodidad de usuario.

Primero de todo, explicar cómo funciona el desplegable o ComboBox, como se llama en VBA. En este caso, como las opciones las da directamente el programa, generar el desplegable es muy sencillo. Simplemente hay que ir a las propiedades del Combobox, y añadir donde se encuentra los datos del listado en RowSource, como podemos ver en la siguiente imagen.

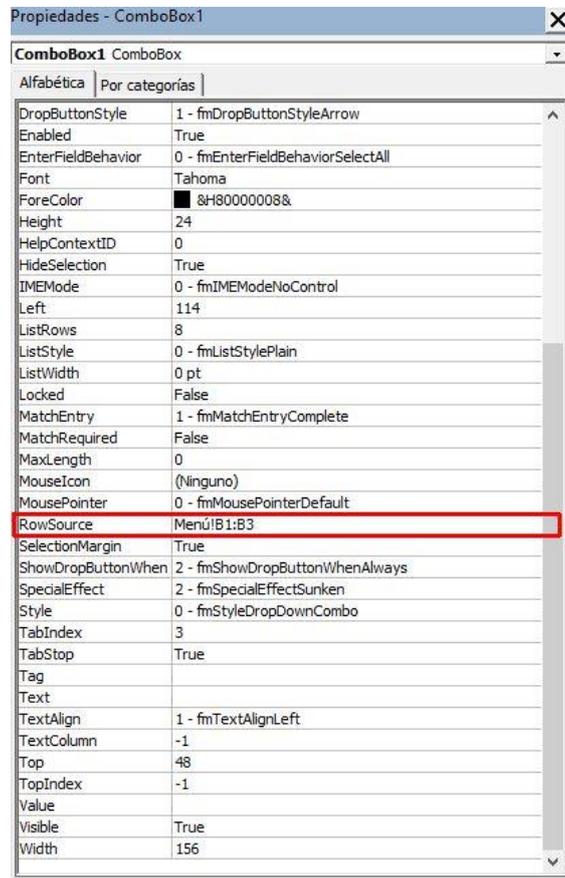


Ilustración 15. Propiedades Combobox Formulario estructura.

En cuanto al código VBA que encontramos bajo el botón de “Añadir sección”:

```
Private Sub CommandButton1_Click()
    Dim fila As Integer
    Dim final As Integer 'Detectar el ultimo registro
    Dim registro As Integer
    Dim pregunta As String
    Hoja2.Activate

    ' 'Recorrer el bucle para detectar cual es la última fila.

    For fila = 1 To 1000
        If Hoja2.Cells(fila, 4) = "" Then

            final = fila
        End If
    Next fila
End Sub
```

```

' Si se añade cualquier otra sección, menos Administración y
Comercial, no deja seleccionar del desplegable la opción
Proporcional

    If (Me.text_sección <> "Administracion" And Me.text_sección
<> "Comercial" And Me.ComboBox1 = "Proporcional") Then

        Me.ComboBox1.BackColor = &H8080FF ' si hay error,
cambia el color de fondo y sale mensaje por pantalla.
        pregunta = MsgBox("No se puede asignar imputación por
proporción a esa sección. Únicamente disponible para Administración
y Comercial.", vbOKOnly, "Error en la selección del tipo de
imputación")

        Exit Sub

    End If

    For registro = 2 To final 'solo recorre lo que tenemos, no
hasta 1000, bucle para ver si hay algun registro repetido
    If ActiveSheet.Cells(registro, 4) = Me.text_sección Then

        Me.text_sección.BackColor = &H8080FF ' si hay error, cambia
color de fondo y sale mensaje por pantalla
        pregunta = MsgBox("Esta sección ya ha sido añadida en la
base de datos. Por favor, intentelo con una distinta.", vbOKOnly,
"Registro de sección duplicado ")
        Me.text_sección.SetFocus
        Exit Sub

    End For

End If
Next

    Hoja2.Cells(final, 4) = Me.text_sección
    Hoja2.Cells(final, 5) = Me.ComboBox1

    'borrar los datos introducidos para poder ingresar nuevos sin
tener que borrar y poner color blanco
    Me.text_sección = ""
    Me.ComboBox1 = ""
    Me.ComboBox1.BackColor = &HFFFFFF
    Me.text_sección.BackColor = &HFFFFFF

End Sub

```

Como se puede apreciar, el código, pese a ser algo más largo que en el anterior formulario, sigue la misma estructura. Únicamente se ha añadido la siguiente sección:

```

If (Me.text_sección <> "Administracion" And Me.text_sección <>
"Comercial" And Me.ComboBox1 = "Proporcional") Then

    Me.ComboBox1.BackColor = &H8080FF ' si hay error,
cambia el color de fondo y sale mensaje por pantalla.

```

```
pregunta = MsgBox("No se puede asignar imputación por proporción a esa sección. Unicamente disponible para Administración y Comercial.", vbOKOnly, "Error en la selección del tipo de imputación")

Exit Sub
```

Con estas líneas de código pretendemos controlar que, si se añade cualquier sección distinta a Administración o Comercial, y seleccionamos en la lista desplegable la opción “Proporcional”, al pulsar en el botón añadir sección nos saltará el siguiente mensaje por pantalla:

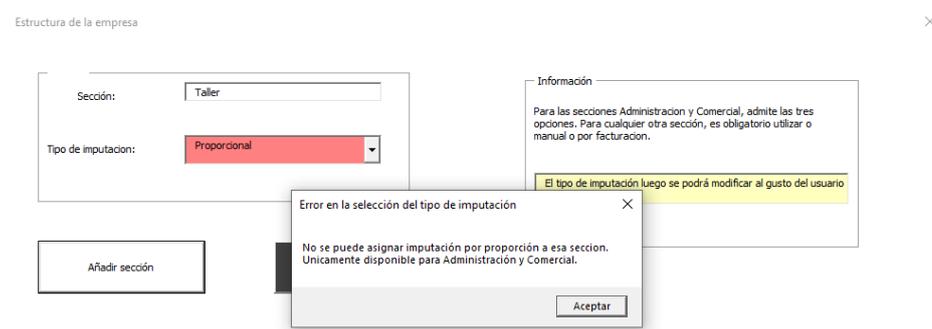


Ilustración 16. Error tipo imputación.

Como se puede ver en la ilustración 16, al querer añadir la sección taller y proporcional, nos sale el mensaje de error y, a mayores, nos pinta de color rojo donde está el error. Este tipo de medidas se presentan en el resto de los formularios.

Formulario Fm_RegistroProductos.

En este formulario podemos registrar las diferencias existencias y unidades producidas y consumidas, así como las finales de los productos finales.

Cuenta con un desplegable que accede a los productos añadidos previamente, y para diferenciar entre productos finales o en curso, se ha añadido un botón de opciones.

Registro de Productos Finales ×

Registro existencia de productos

Nombre:

Existencias iniciales. €

Cantidad (Uds) producida:

Cantidad (Uds) consumida

Unidades Finales:

Selección tipo de producto

Producto Terminado

Producto en curso

Información

Para productos en curso, no se puede añadir cantidad producida ni consumida.

Ilustración 17. Formulario Registro de productos.

En este formulario se ha utilizado una estrategia distinta, ya que, al crear el botón de opciones, generaba problemas a la hora de programarlo directamente en el botón de registrar producto. Así que, para solucionar esto se optó por hacer una macro aparte, llamada RegistrarProductos. Una vez dentro del botón Registrar Producto, haremos una llamada a esta macro y funcionará como cualquier otro formulario. Veamos el código de esta macro primero, antes de explicar el código bajo el botón de registro de producto.

```

Sub RegistrarProductos ()
    Dim fila As Integer
    Dim final As Integer 'Detectar el ultimo registro
    Dim registro As Integer

    For fila = 2 To 1000
        If ActiveSheet.Cells(fila, 2) = "" Then

                final = fila
            Exit For
        End If
    Next

    For registro = 2 To final 'solo recorre lo que tenemos, no
hasta 1000, bucle para ver si hay algun registro repetido
        If ActiveSheet.Cells(registro, 2) =
FM_RegistroProductos.ComboBox1 Then
            MsgBox "Registro ya existente"
            Exit Sub

        Exit For
    End If
Next
    
```

```

ActiveSheet.Cells(final, 1) = fila - 1
'Pegamos registros en la hoja correspondiente
ActiveSheet.Cells(final, 2) = FM_RegistroProductos.ComboBox1
ActiveSheet.Cells(final, 3) =
Val(FM_RegistroProductos.text_iniciales)
ActiveSheet.Cells(final, 4) =
Val(FM_RegistroProductos.text_cantidadconsumida)
ActiveSheet.Cells(final, 5) =
Val(FM_RegistroProductos.text_cantidadvendida)
ActiveSheet.Cells(final, 8) =
Val(FM_RegistroProductos.Text_unidadesfinales)
'borrar los datos introducidos para poder ingresar nuevos sin
tener que borrar
FM_RegistroProductos.ComboBox1 = ""
FM_RegistroProductos.text_iniciales = ""
FM_RegistroProductos.text_cantidadvendida = ""
FM_RegistroProductos.Text_unidadesfinales = ""
End Sub

```

Como se puede ver, el código es muy similar a lo ya utilizado con anterioridad, únicamente hemos tenido que utilizar el comando `ActiveSheet.Cells` ya que, depende de la opción que se elija (botón de opciones) tendrá que copiarlo en una u otra hoja.

En cuanto al código del formulario, vemos que queda muy simple:

```

Private Sub boton_Registrar_Click()

    If Me.opt_PT.Value = True Then 'segun la opcion que esté
    marcada en el formulario activa una u otra hoja
        Hoja3.Select

    Else
        Hoja4.Select

    End If
    'Hacemos llamada a la macro que recoge los datos del
    formulario

    Call RegistrarProductos

End Sub

```

Como se ha utilizado un mismo formulario para registrar tanto los productos finales como los en curso, y estos últimos solo tiene sentido anotar las existencias que hay en curso, encontramos la problemática de que se quedaban muchos textbox sin rellenar. Para solucionar esto, generamos un pequeño código que, si se activa la opción `Productos en curso`, se desactivan los textbox que no son necesario. Veamos.

Registro de Productos Finales ×

Registro existencia de productos

Nombre:

Existencias iniciales. €

Selección tipo de producto

Producto Terminado

Producto en curso

Información

Para productos en curso, no se puede añadir cantidad producida ni consumida.

Ilustración 18. Formulario Registro productos en curso.

Como se puede ver en la ilustración 18, al tener marcada la opción Producto Terminado, el formulario presentaba varias opciones de registro más. Al seleccionar la segunda opción, vemos que nos quedamos únicamente con el desplegable de producto y las existencias iniciales.

Añadimos el código con el que se consigue esto.

```
Private Sub op_PC_Change ()
'Segun marcamos una u otra opcion, determinamos si el textbox
esta visible o no.
Me.text_cantidadconsumida.Visible = Not op_PC.Value

Me.CantidadProd.Visible = Not op_PC.Value

Me.text_cantidadvendida.Visible = Not op_PC.Value

Me.CantidadCon.Visible = Not op_PC.Value

Me.Tex_unidadesfinales.Visible = Not op_PC.Value
Me.Unidadesfinales.Visible = Not op_PC.Value
End Sub
```

Con esta subrutina, conseguimos el objetivo deseado y ya tenemos dos formularios distintos, según el botón marcado.

Y, por último, explicar cómo funciona el Combobox en este caso, ya que, en el caso anterior, veíamos como simplemente era editando las propiedades del mismo, pero como en este caso el número de opciones del listado son ilimitadas, tenemos que crear una subrutina que acceda a estos datos y los traiga al desplegable.

```
Private Sub ComboBox1_Enter()  
    Dim fila As Integer  
    Dim final As Integer  
    Dim listado As String  
  
    For fila = 1 To ComboBox1.ListCount  
        ComboBox1.RemoveItem 0  
    Next fila  
  
    For fila = 2 To 1000 ' calculamos el tamaño de filas que hay y  
        lo acumulamos en la variable final  
        If Hoja2.Cells(fila, 7) = "" Then  
            final = fila - 1  
            Exit For  
        End If  
    Next  
  
    For fila = 2 To final 'añadimos al listado los articulos que  
        encuentre segun la hoja activa (opciones)  
  
        listado = Hoja2.Cells(fila, 7)  
        ComboBox1.AddItem (listado)  
    Next  
  
End Sub
```

Con estas líneas, generamos el desplegable correspondiente, en este caso, accede a la Hoja2, hasta la última fila que detecte el bucle, columna G.

Este código se reutiliza en todos los Combobox que tengan los formularios, editando, como es obvio, donde accede a los datos del listado.

Formulario FM_existenciasiniciales.

Con este módulo registraremos las distintas materias primas, así como las cantidades iniciales que tenemos de las mismas. Por último, añadiremos las existencias finales que tenemos.

Ilustración 19. Formulario Existencias iniciales Materias primas

En cuanto al formulario, sigue la línea de los módulos anteriores. Como novedad, incluye un pequeño calendario para añadir la fecha en la cual se tiene constancia de la existencia de esas existencias.

El código VBA de este formulario es el siguiente:

```
Private Sub CommandButton1_Click()
    Dim fila As Integer
    Dim final As Integer 'Detectar el ultimo registro

    Hoja5.Activate

    For fila = 2 To 1000
        If Hoja5.Cells(fila, 2) = "" Then
            final = fila
            Exit For
        End If
    Next
    Hoja5.Cells(final, 1) = "Inicial"
    Hoja5.Cells(final, 2) = Me.text_nomexistencia
    Hoja5.Cells(final, 3) = Me.texto_fecha

    Hoja5.Cells(final, 4) = Val(Me.text_cantidad)
    Hoja5.Cells(final, 6) = Val(Me.text_total)
    Hoja5.Cells(final, 5) = Me.text_total / Me.text_cantidad
    Hoja5.Cells(final, 7) = Val(Me.Text_finales)
    'borrar los datos introducidos para poder ingresar nuevos sin
    tener que borrar
    Me.text_nomexistencia = ""
    Me.text_cantidad = ""    Me.text_total = ""
    Me.Text_finales = ""
End Sub
```

Como novedad, en este formulario se añade esta línea de código

```
Hoja5.Cells(fila, 5) = Me.text_total / Me.text_cantidad
```

Con esta línea, hacemos el cálculo que nos cuesta por unidad la materia prima, sin registrarla. El resto del código ya es conocido en otros formularios, por lo que no requiere explicación.

Formulario FM_Compas.

En este apartado tenemos que hablar del formulario que recoge todas las compras de materias primas que se realizan. Permite añadir la fecha de las mismas y hace el cálculo del precio por unidad de cada una.

Ilustración 20. Formulario Compra Materias primas

Como se puede ver, se incluye un Combobox, que recoge los datos de materias primas introducidas con el formulario de existencias iniciales de materias primas.

El código del mismo:

```
Private Sub Combobox1_compras_Enter()
    Dim fila As Integer
    Dim final As Integer
    Dim listado As String

    For fila = 1 To ComboBox1_compras.ListCount
        ComboBox1_compras.RemoveItem 0
    Next fila

    For fila = 2 To 1000 ' calculamos el tamaño de filas
        que hay y lo acumulamos en la variable final
        If Hoja5.Cells(fila, 2) = "" Then
            final = fila - 1
        Exit For
    End If
```

```
Next

    For fila = 2 To final 'añadimos al listado los artículos
que encuentre segun la hoja activa (opciones)

        listado = Hoja5.Cells(fila, 2)
        ComboBox1_compras.AddItem (listado)
    Next
End Sub
```

Y en cuanto al código VBA para registrar cada compra de materia prima:

```
Private Sub CommandButton1_Click()
    Dim fila As Integer
    Dim final As Integer 'Detectar el ultimo registro
    Dim registro As Integer

    Hoja17.Activate

    For fila = 2 To 1000
        If Hoja17.Cells(fila, 2) = "" Then

                final = fila
            Exit For
        End If
    Next

    Hoja17.Cells(final, 1) = "Compra"

    Hoja17.Cells(final, 2) = Me.ComboBox1_compras

    Hoja17.Cells(final, 3) = Me.texto_fecha compra
    Hoja17.Cells(final, 4) = Val(Me.Texto_cantidad compra)

    Hoja17.Cells(final, 6) = Val(Me.texto_precio compra)
    Hoja17.Cells(final, 5) = Me.texto_precio compra /
Me.Texto_cantidad compra

End Sub
```

Este código no presenta ninguna novedad respecto a los ya explicados anteriormente.

Formulario FM_ConsumoMP.

Ocurre lo mismo con este nuevo formulario. En él registraremos la salidas del almacén o consumos de materias primas. Es totalmente igual al formulario de compras, cambiando las variables a cada etiqueta, hojas... etcétera. Pero en esencia son lo mismo.

Ilustración 21. Formulario Consumo Materias Primas

El código para añadir el consumo:

```
Private Sub CommandButton1_Click()
    Dim fila As Integer
    Dim final As Integer 'Detectar el ultimo registro
    Dim registro As Integer
    Dim pregunta As String

    Hoja18.Activate
    For fila = 2 To 1000
        If Hoja18.Cells(fila, 2) = "" Then

            final = fila
            Exit For
        End If
    Next

    Hoja18.Cells(final, 1) = "consumo"

    Hoja18.Cells(final, 2) = Me.ComboBox1_consumo

    Hoja18.Cells(final, 3) = Me.fecha_consumo
    Hoja18.Cells(final, 4) = Val(Me.Texto_cantidadconsumo)
End Sub
```

Formulario FM_Personal.

Con este formulario queremos controlar todos los salarios de los trabajadores. Con él, pretendemos añadir el nombre del trabajador, sección en la que trabaja, y para tener un mayor control, se ha añadido las fechas de inicio y fin del

contrato, dando mucho más realismo a la herramienta. Luego se calcula el número de días trabajados y el salario real que cuesta el trabajador.

Las líneas de código para añadir cada registro es el siguiente:

```
Private Sub CommandButton1_Click()  
    Dim fila As Integer  
    Dim final As Integer 'Detectar el ultimo registro  
    Dim registro As Integer  
    Dim pregunta As String  
    For fila = 2 To 1000  
        If Hoja6.Cells(fila, 2) = "" Then  
            final = fila  
            Exit For  
        End If  
    Next  
  
    For registro = 2 To final 'solo recorre lo que tenemos, no  
hasta 1000, bucle para ver si hay algun registro repetido  
    If ActiveSheet.Cells(registro, 2) = Me.text_nompersonal Then  
  
        Me.text_nompersonal.BackColor = &H8080FF ' si hay  
error, cambiar color  
        pregunta = MsgBox("Este trabajador ya se encuentra en  
la base de datos. Pruebe con otro.", vbOKOnly, " Error  
al añadir trabajador")  
        Me.text_nompersonal.SetFocus  
        Exit Sub  
  
        Exit For  
    End If  
Next  
    'Copiamos todo lo introduccido en el formulario  
    Hoja6.Cells(final, 2) = Me.text_nompersonal  
    Hoja6.Cells(final, 3) = Me.text_fecha  
    Hoja6.Cells(final, 4) = Val(Me.text_salario)  
    Hoja6.Cells(final, 5) = Me.ComboBox1  
    Hoja6.Cells(final, 6) = Me.text_fechafin  
    Hoja6.Cells(final, 7) = Me.text_fechafin - Me.text_fecha  
    'Calcula diferencia de dias trabajados  
    Hoja6.Cells(final, 8) = (Me.text_fechafin - Me.text_fecha) / (365)  
    'Proporcion de dias trabajados  
    Hoja6.Cells(final, 9) = ((Me.text_fechafin - Me.text_fecha) /  
(365)) * Me.text_salario ' Salario Real  
  
    'borrar los datos introducidos para poder ingresar nuevos sin  
tener que borrar  
  
    Me.text_nompersonal = ""  
    Me.text_fecha = "01/01/2020"  
    Me.text_salario = ""  
    Me.ComboBox1 = ""  
    Me.text_nompersonal.BackColor = &HFFFFFF  
End Sub
```

En este código presentamos alguna novedad, como puede ser mensajes de error si el trabajador se repite.

```
For registro = 2 To final 'solo recorre lo que tenemos, no hasta 1000, bucle para ver si hay algun registro repetido
    If ActiveSheet.Cells(registro, 2) = Me.text_nompersonal
Then
    Me.text_nompersonal.BackColor = &H8080FF ' si hay error, cambiar color
    pregunta = MsgBox("Este trabajador ya se encuentra en la base de datos. Pruebe con otro.", vbOKOnly, " Error al añadir trabajador")
    Me.text_nompersonal.SetFocus
    Exit Sub

    Exit For
End If
Next
```

Otra curiosidad es el cálculo del salario real, a nivel de coste, que se realiza cada vez que registramos un trabajador.

```
Hoja6.Cells(final, 7) = Me.text_fechafin - Me.text_fecha 'Calcula diferencia de dias trabajados
Hoja6.Cells(final, 8) = (Me.text_fechafin - Me.text_fecha) / (365) 'Proporcion de dias trabajados
Hoja6.Cells(final, 9) = ((Me.text_fechafin - Me.text_fecha) / (365)) * Me.text_salario ' Salario Real
```

Es en estas líneas donde se realizan estos cálculos, primero calcula la diferencia de fechas, luego la proporción, y finalmente lo multiplica por el salario anual del trabajador.

Formulario FM_Facturacion.

Con este formulario controlamos las ventas por producto de la compañía. Es un formulario muy sencillo que nos permite añadir, producto a producto la facturación del mismo.

Ilustración 22. Formulario Facturación

Como se puede apreciar en la ilustración 22, es uno de los formularios más sencillos. En cuanto al código del mismo es el siguiente:

```
Private Sub CommandButton1_Click()
    Dim fila As Integer
    Dim final As Integer 'Detectar el ultimo registro
    Dim registro As Integer
    Dim pregunta As String
    For fila = 2 To 1000
        If Hoja10.Cells(fila, 2) = "" Then

            final = fila
            Exit For
        End If
    Next

    For registro = 2 To final 'solo recorre lo que tenemos, no
hasta 1000, bucle para ver si hay algun registro repetido
        If ActiveSheet.Cells(registro, 2) = Me.ComboBox1 Then

            Me.ComboBox1.BackColor = &H8080FF
            pregunta = MsgBox("Este producto ya tiene facturación, si
lo desea puede modificar este valor.", vbOKOnly, "Registro de
facturación duplicado ")
            Me.ComboBox1.SetFocus
            Exit Sub
        End If
    Next

    Hoja10.Cells(final, 2) = Me.ComboBox1

    Hoja10.Cells(final, 3) = Val(Me.text_facturacion)
    'borrar los datos introducidos para poder ingresar nuevos
sin tener que borrar
    Me.text_facturacion = ""
    Me.ComboBox1 = ""
    Me.ComboBox1.BackColor = &HFFFFFF
End Sub
```

Formulario FM_Gastos.

Ya, por último, para finalizar de mencionar los formularios de la herramienta, tenemos que hablar del formulario de gastos.

Con este formulario queremos registrar todos los gastos que ocurran en la compañía (luz, internet, amortizaciones...etcétera.)

The screenshot shows a window titled "Gastos generales de la empresa" with a close button (X) in the top right corner. The form contains the following elements:

- Nombre:** A text input field.
- Fecha inicio:** A date dropdown menu showing "01/01/2020".
- Fecha final:** A date dropdown menu showing "31/12/2020".
- Gasto:** A text input field.
- Departamento:** A dropdown menu.
- NOTA:** A text box containing the text: "Si un gasto se imputa a varias secciones, se deberá añadir el importe por cada sección, no el total. Puede utilizar la calculadora usando el botón de abajo." Below the text is a light gray rectangular area.
- Buttons:** Three buttons at the bottom: "Añadir gasto" (white), "Cancelar" (black), and "Abrir calculadora" (green).

Ilustración 23. Formulario Gastos.

Como se puede apreciar en la ilustración anterior, tenemos un formulario que nos permite ingresar cada coste individualmente por sección. Y como esto pudiera presentar algún problema, ya que quizá tenemos el gasto total y queremos repartirlo en varias secciones, se ha añadido un botón que ejecuta la calculadora de Windows, con el objetivo de dar facilidades al usuario final.

Para ejecutar esta calculadora hemos necesitado el siguiente código, sacado de (Muñoz, 2019):

```
Private Sub CommandButton22_Click ()
    Application.ActivateMicrosoftApp Index:=0
End Sub
```

Por otro lado, el código del formulario es el siguiente, muy similar al de personal, mencionado con anterioridad.

```
Private Sub CommandButton21_Click ()
    Dim fila As Integer
    Dim final As Integer 'Detectar el ultimo registro

    For fila = 2 To 1000
        If Hoja7.Cells(fila, 2) = "" Then
```

```
                final = fila
            Exit For
        End If
    Next

        Hoja7.Cells(final, 2) = (fila - 1) & "-"
    & Me.text_nomgasto
        Hoja7.Cells(final, 3) = Me.text_fechagasto
        Hoja7.Cells(final, 4) = Val(Me.text_gasto)
        Hoja7.Cells(final, 5) = Me.ComboBox1
        Hoja7.Cells(final, 6) = Me.text_fechafin
        Hoja7.Cells(final, 7) = Me.text_fechafin
    - Me.text_fechagasto
        Hoja7.Cells(final, 8) = (Me.text_fechafin
    - Me.text_fechagasto) / (365)
        Hoja7.Cells(final, 9) = ((Me.text_fechafin
    - Me.text_fechagasto) / (365)) * Me.text_gasto

        'borrar los datos introducidos para poder ingresar
nuevos sin tener que borrar

        Me.text_nomgasto = ""
        Me.text_fechagasto = "01/01/2020"
        Me.text_gasto = ""
        Me.ComboBox1 = ""

End Sub
```

En este formulario se añade, de nuevo, la capacidad de calcular el gasto real entre dos fechas, por si un gasto no se contrata al completo durante el año.

Módulos.

En esta sección pasaremos a hablar de los módulos que se encuentran en la aplicación. Es la parte del desarrollo del proyecto más compleja, ya que ha requerido varios reajustes para ir añadiendo más y más funciones. Es por esto por lo que se tomó la decisión de hacerlo en diferentes módulos, y no todo en uno genérico, ya que hubiesen quedado demasiado largos y el haber localizado errores hubiera sido mucho más complicado.

Para facilitar la comprensión del lector, no vamos a incluir todas las macros que salen en la ilustración 10, ya que las que son nombradas como “lanzar_” y “ver_” son iguales y solo es interesante ver un par por cada grupo.

- Macros grupo 1: Aquí hablaremos de las más importantes, las cuales realizan todos los cálculos necesarios para obtener los resultados de contabilidad de costes, así como borrar todos los datos. El código

completo de estos módulos se ha añadido en la sección auxiliar de anexos, al final del documento.

- Macros grupo 2: Mostraremos las macros que sirven para lanzar los formularios.
- Macros grupo 3: Hablaremos de las macros que sirven para desplazarnos entre las distintas hojas del programa.

Macros grupo 1.

Módulo A1_Hoja_Analitica_secciones.

El objetivo de esta macro es generar la Hoja analítica de secciones, separando los gastos directos e indirectos y repartiéndolo en secciones.

Como objetivo secundario, genera la página configurador, donde se tiene que añadir la cantidad que se imputa por producto, o incluso cambiar el tipo de imputación en cada una de las secciones.

Esta macro fue de las más complicadas de obtener, no por su dificultad, sino porque muchas de las líneas de este código han sido reutilizadas en las siguientes macros. Como por ejemplo toda vez que se ha necesitado incluir una fórmula y arrastrarla, como en este caso:

```
With Range("c4")
    .Formula
    "=iferror(vlookup(concatenate($b4,c$3),personal!$a:$i,9,false),iferror(vl
ookup(concatenate($b4,c$3),gastos!$a:$i,9,false),"-"))"

    .AutoFill .Resize(, .Offset(-1).End(xlToRight).Column - .Column + 1),
xlFillDefault

With .Resize(, .Offset(-1).End(xlToRight).Column - .Column + 1)
    .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row + 1),
xlFillDefault
End With

End With
```

Muy útil durante toda la implementación de la herramienta, con sus ajustes correspondientes.

Otro código muy importante a destacar, que se ha utilizado en numerosas ocasiones ha sido el de sumar columnas, que se hizo por primera vez en esta macro:

```
With Hoja11
    fila = .Range("C" & Rows.Count).End(xlUp).Row
    ' Última Fila ocupada
    columna = .Cells(3,
Cells.Columns.Count).End(xlToLeft).Column ' Última Columna ocupada
```

```
Set d = .Range(.Cells(4, 3), .Cells(4, columna)) 'Crea un Rango de la primera fila
For Each celda In d ' Paso por todas las celdas del rango de la primera fila
    Set c = .Range(Cells(3, celda.Column), Cells(fila, celda.Column)) ' creo un rango con la columna a sumar
    .Cells(fila + 1, celda.Column) = Application.WorksheetFunction.Sum(c) ' sumo el rango con la funcion Sum
    Set c = Nothing ' vacio por si acaso
Next celda
End With

End With
```

A modo resumen, hay que indicar que primero busca tanto las secciones como los distintos costes (personal y gastos). Una vez tiene los ejes de la tabla creados, busca en las distintas hojas el gasto y la sección.

Para ello nos valemos de la siguiente formula:

```
"=iferror(vlookup(concatenate($b4,c$3),personal!$a:$i,9,false),iferror(vlookup(concatenate($b4,c$3),gastos!$a:$i,9,false),"-"))"
```

Esta fórmula no es más que un SI.ERROR(BUSCARV ()) que accede a tanto la hoja de personal o a la de gastos.

Para saber en qué celda debe añadirlo, concatenamos la palabra del gasto con la sección en la columna A, haciendo lo mismo en las hojas de personal y gasto.

[Módulo A2_Imputacion_por_Productos.](#)

Con esta macro recogemos los costes calculados en la hoja analítica por secciones, y junto con costes de productos en curso y productos finales, repartirlo, según la imputación elegida en el Configurador.

Una vez tenemos el coste industrial, podemos determinar el coste unitario por producto, y a su vez, calcular el consumo real en el periodo contable.

[Módulo A3_Cuadro_Margenes .](#)

El resultado de esta macro es calcular el último cuadro de la contabilidad de costes, generando el Cuadro de márgenes y resultados.

Para ello, recoge la facturación por producto, el coste industrial, y calcula los distintos márgenes (margen industrial, comercial y administrativo) hasta llegar al resultado de la actividad. Es en este punto donde la herramienta accederá a la hoja de ajustes y, una vez los tiene en cuenta, presenta el Resultado del Periodo, todo ello calculado por producto.

Módulo A4_Almacén.

Con esta macro se tiene como objetivo principal la gestión del almacén, realizando el método PMP.

Antes de nada, necesitamos la facturación, por lo cual, si no la ha añadido nos pedirá que lo hagamos. El programa accede a las distintas hojas de existencias, compras y salidas y las pegará en la hoja donde se va a realizar el PMP. Una vez tenemos todas las materias pegadas, ordenará esa lista por materia prima y por fecha. Y una vez tenga ordenada la tabla, ingresa las fórmulas en las celdas del arriba , para luego ser arrastradas hasta el final.

Cuando tenemos ya los consumos totales, por facilitar el código, creamos una hoja auxiliar, donde sacamos los consumos totales por cada materia prima.

A continuación, añadimos una fila vacía de separación entre materias primas dentro de la hoja Almacén.

Por último, generamos y completamos la hoja de existencias con las diferencias, si las hubiera.

Macros grupo 2.

Como he comentado anteriormente, en este punto solo mostraré un par de macros, ya que son exactamente iguales.

Módulo Lanzar_Compras

```
Sub lanzarcompra(registro As IRibbonControl)
    Hoja17.Activate
    FM_Compras.Show
End Sub
```

Como podemos ver, esta macro tiene como argumento a la función registro As IRibbonControl, que hemos visto y explicado con anterioridad. Esto indica que es un botón en la cinta de opciones. Si analizamos el código, vemos que activa la hoja 17 (Compras) y muestra el formulario de compras, obviamente.

Módulo Lanzar_Personal.

```
Sub lanzar_fm_personal(formulario1 As IRibbonControl)
    Hoja6.Activate
    FM_Personal.Show
End Sub
```

En la imagen anterior podemos ver que el código del programa es igual al anterior. En este caso, se abriría la hoja 6 (Personal) y nos abriría el formulario de Personal.

Macros grupo 3.

Como ocurre en el punto anterior, solo se mostrarán dos macros a modo de muestra, ya que las siguientes son exactamente iguales.

Módulo ver_CompasMP.

En este caso, y para comparar, veremos el botón que usaremos para consultar la bases de datos y movernos entre hojas. En este caso, a la hoja de compras de materias primas.

```
Sub comprasmp(registro As IRibbonControl)
    Hoja17.Activate
End Sub
```

Es una macro muy simple, como se puede ver mantiene el argumento a la función que la convierte en botón en la cinta de opciones. Esta macro, únicamente nos abre la hoja17 (Compras).

Módulo Ver_Personal.

```
Sub bdpersonal(registro As IRibbonControl)
    Hoja6.Activate
End Sub
```

La estructura del código es la misma, únicamente cambia, como es obvio, la página que abre al pulsar el botón. En este caso, nos abriría la hoja6 (Personal).

CAPÍTULO 3: MANUAL DE USUARIO

En este capítulo del trabajo, se pretende exponer como funciona al completo la herramienta. Para ello, se resolverá un ejercicio utilizando la herramienta diseñada.

Enunciado del problema.

La empresa CUCHILLAS se dedica a la fabricación y comercialización de cuchillas de afeitar de usar y tirar. Su proceso productivo lo tiene organizado de la siguiente forma:

- Tres secciones productivas: Laminado, Troceado y Tratamiento.
- Una sección auxiliar.
- Una sección administrativa.

La materia prima (fleje de acero), es valorada mediante el método del Precio Medio Ponderado, disponiéndose de los siguientes datos:

Existencias	Iniciales	Finales
Fleje de acero	1.300 kg a 10 €/kg	38.000 €
Cuchillas de afeitar en curso	3000 €	984 €
Cuchillas de afeitar Terminadas	0	600.000 €

A lo largo del periodo:

- Se han comprado 45.000 kg. de fleje de acero por importe de 410.000 €, y se ha contratado, además, su transporte por importe de 16.850 €.
- Se ha consumido 42.000 kg. de fleje de acero.
- Y se han fabricado, en términos físicos, 5.374.000 cuchillas de afeitar terminadas.

El resto de los costes del periodo son los siguientes:

- Personal directo: 12 operarios con un coste de 34.375 €/operario
- Costes indirectos:
 - Laminado: 203.180 €
 - Troceado: 108.090 €
 - Tratamiento: 140.950 €
 - Auxiliar: 112.140 €

- Administración: 167.140 €

Información adicional:

- Durante el periodo se han vendido 4.750.000 cuchillas de afeitar a 0,35 €/unidad.

[Página principal.](#)

Cuando iniciamos el programa nos encontramos con la siguiente imagen.

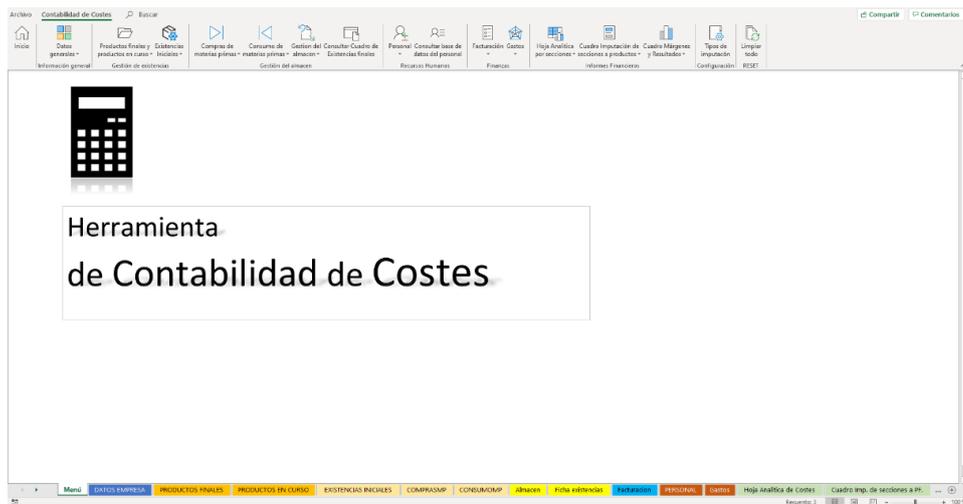


Ilustración 24. Menú principal

Encontramos una interfaz muy limpia, con todos los botones en la cinta de opciones, aunque si el usuario lo prefiere, puede moverse por las hojas de Excel de manera tradicional.

Una vez visto el menú principal, al cual podemos volver siempre que queramos pulsando en el botón de la cinta de opciones Inicio, pasamos a introducir los datos del ejercicio.

[Datos de la empresa.](#)

Comenzamos añadiendo los distintos datos conocidos de la empresa, así como los productos de la misma. Para ello pulsamos en el botón “Datos generales” y, a continuación, damos al botón “Información”.

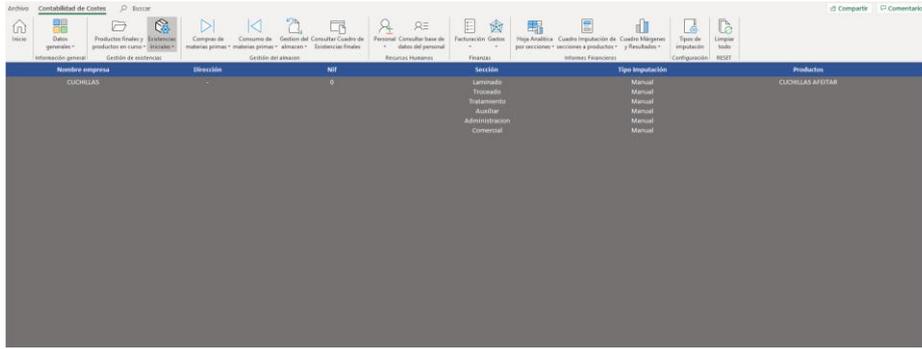


Ilustración 27. Datos Empresa completo

Como se puede ver, pese que el enunciado no nos menciona la sección Comercial, nosotros la añadimos, ya que luego realizará el margen comercial, y así nos aseguramos de que el resultado final es el correcto.

Productos finales y en curso.

En este punto tenemos que añadir los productos finales y en curso que tenemos en la producción. Para ello pulsamos en Productos finales y Productos en Curso → Añadir Productos finales y en curso.

Mediante el siguiente formulario rellenamos la información que nos dice el enunciado. Primero añadimos los productos finales

The screenshot shows a 'Registro de Productos Finales' form. It has a title bar with a close button (X). The form is titled 'Registro existencia de productos' and contains the following fields and options:

- Nombre:** Dropdown menu with 'CUCHILLAS AFEITAR' selected.
- Existencias iniciales. €:** Text input field with '0'.
- Cantidad (Uds) producida:** Text input field with '5374000'.
- Cantidad (Uds) consumida:** Text input field with '4750000'.
- Unidades Finales:** Text input field with '600000'.
- Selección tipo de producto:** Radio buttons for 'Producto Terminado' (selected) and 'Producto en curso'.
- Información:** A green box containing the text: 'Para productos en curso, no se puede añadir cantidad producida ni consumida.'
- Buttons:** 'Registrar Producto' and 'Cancelar'.

Ilustración 28. Formulario Registro Producto Terminado

Una vez pulsamos en Registrar productos, vemos que en pantalla tenemos lo siguiente.

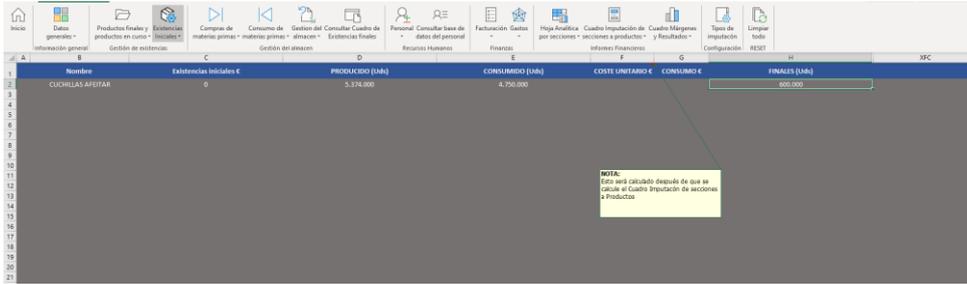


Ilustración 29. Imagen Productos en curso en pantalla.

Continuamos con los productos en curso, para ello simplemente abrimos el mismo formulario y seleccionamos “Producto en curso” en el selector de tipo de producto.



Ilustración 30. Formulario Registro Productos en curso.

Como vemos y ya explicamos en la Guía del programador, el formulario cambia, dejando únicamente las existencias iniciales, ya que son las que se van a consumir. Añadimos las existencias pulsando el botón, obteniendo lo siguiente:

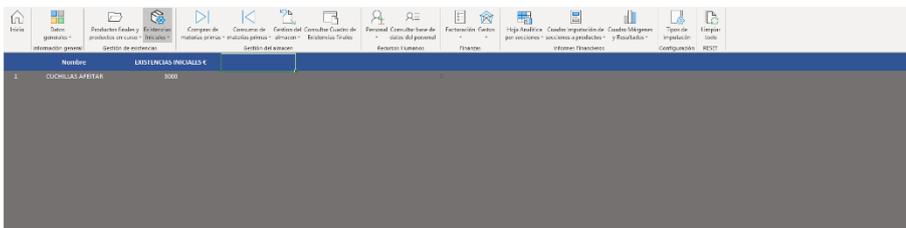


Ilustración 31. Imagen Productos en curso en pantalla

Existencias iniciales de materias primas.

Continuando con el ejercicio, debemos añadir las existencias iniciales de materia prima. En este caso, flejes de acero.

Para ello, utilizamos el cuarto botón de la cinta de opciones: Existencias iniciales → Existencias

Existencias iniciales de Materias Primas

Registro de materias primas

Nombre: Flejes de acero

Fecha: 01/01/2020

Cantidad: 1300 Uds precio: 13000

Existencias Finales

Cantidad: 38000

Añadir Existencias Cancelar

Ilustración 32. Formulario Existencias iniciales materia prima

Nombre	Fecha	Cantidad	Precio	Total	Existencias Finales	
Inicial	Flejes de acero	01/01/2020	1300	13,000 €	13,000,00 €	38000

Ilustración 33. Imagen Existencias iniciales materia prima

Compras de materias primas.

Una vez tenemos completadas las existencias iniciales de la materia prima, podemos ingresar las materias primas compradas y consumidas. Comenzamos hablando de las materias primas que se compran en el periodo contable. Para ello nos vamos a Compras de materias primas y nos lanza el correspondiente formulario, el cual rellenamos con los datos del enunciado:

Compra de materias primas X

Registro de compras de materias primas

Nombre:

Fecha:

Cantidad: Precio: €

Ilustración 34. Formulario compra materias prima

El enunciado como tal no nos indica la fecha que se realiza la compra, asique ponemos cualquiera, siempre y cuando tenga sentido.

CONCEPTO	MATERIA PRIMA	FECHA	CANTIDAD	PRECIO	TOTAL
Compra	Flejes de acero	28/02/2020	45000	9.49 €	426.850,00 €

Ilustración 35. Imagen Compra materias primas en pantalla.

Consumos de materias primas .

Realizamos lo mismo que en el apartado anterior, únicamente con los consumos de la materia prima.

Consumo materias primas X

Registro de consumo de materias primas

Nombre:

Fecha:

Cantidad:

Ilustración 36. Formulario Consumo Materia prima

Como se aprecia en la imagen anterior, el formulario no nos da la opción de añadir el precio de estos consumos, ya que esto será calculado por el método PMP en la gestión del almacén. Únicamente añadimos la fecha, la cantidad y le damos a añadir consumo.

CONCEPTO	MATERIA PRIMA	FECHA	CANTIDAD
consumo	Flejes de acero	30/12/2020	42000

Ilustración 37. Consumo Materia prima en pantalla.

Facturación.

Añadimos la facturación obtenida durante el periodo, para ello, pulsamos en el botón de Facturación → Facturación.

Con los datos que nos da el enunciado y lo que nos ofrece el formulario de la herramienta, debemos realizar el cálculo de lo facturado por producto, previamente a ingresarlo en el formulario.

Ilustración 38. Formulario Facturación

Producto	Cantidad	1.662.500,00 €
CUCHILLAS AFEITAR		1.662.500,00 €

Ilustración 39. Imagen facturación en pantalla.

Personal.

El enunciado nos dice que tenemos 12 operarios con un sueldo de 34.375€ cada uno. Con lo cual, por cada sección operativa tendríamos un coste de 137.500€. Como no nos indican fecha, suponemos que están contratados durante todo el año. Lo introducimos al programa mediante el botón Personal.

Ilustración 40. Formulario Personal

Una vez añadimos los trabajadores de las tres secciones operativas, obtenemos esto por pantalla:

Nombre	Fecha ingreso	Salario	Departamento	Fecha despidio	Dias contratado en el año	Proporcion dias trabajados	Salario real
Trabajadores Laminado	01/01/2020	137.500,00 €	Laminado	31/12/2020	365	100%	137.500,00 €
Trabajador Troceado	01/01/2020	137.500,00 €	Troceado	31/12/2020	365	100%	137.500,00 €
Trabajador tratamiento	01/01/2020	137.500,00 €	Tratamiento	31/12/2020	365	100%	137.500,00 €

Ilustración 41. Imagen Personal en pantalla.

Gastos.

Ahora debemos añadir los costes indirectos en la empresa. El mismo enunciado nos los da directamente. Los añadimos con el formulario, pulsando en el botón Gastos.

Gastos generales de la empresa ✕

Nombre:

Fecha inicio:

Fecha final:

Gasto:

Departamento:

NOTA:
Si un gasto se imputa a varias secciones, se deberá añadir el importe por cada sección, no el total. Puede utilizar la calculadora usando el botón de abajo.

Ilustración 42. Formulario gastos.

Añadimos todos los gastos indirectos:

Gasto	Fecha inicio	Finca	Departamento	Fecha fin	Días gasto en el año	Proporción días gastados	Gasto real
1-Coste indirecto Laminado	01/01/2020	205.385,00 €	Laminado	31/12/2020	365	100%	205.385,00 €
2-Coste indirecto Troceado	01/01/2020	198.098,00 €	Troceado	31/12/2020	365	100%	198.098,00 €
3-Coste indirecto Tratamiento	01/01/2020	140.350,00 €	Tratamiento	31/12/2020	365	100%	140.350,00 €
4-Coste indirecto Auxiliar	01/01/2020	112.340,00 €	Auxiliar	31/12/2020	365	100%	112.340,00 €
5-Coste indirecto Administración	01/01/2020	167.340,00 €	Administración	31/12/2020	365	100%	167.340,00 €

Ilustración 43. Imagen Gastos en pantalla.

Hoja analítica por secciones y el Configurador Manual.

Una vez tenemos los gastos directos e indirectos introducidos en la aplicación es momento de generar informes económicos. Comenzamos por la Hoja Analítica por secciones, donde repartirá todos los gastos por secciones. Una vez los tengamos, tendremos que acudir al Configurador de la aplicación para confirmar y determinar cómo se imputan por productos los distintos costes. En este caso, todo irá para el mismo producto, pero si tuviéramos varios productos podríamos elegir el porcentaje concreto a imputar por producto, o si bien, por facturación o de manera proporcional.

Pulsamos el botón Hoja analítica por secciones → Generar cuadro.

Automáticamente nos lleva al configurador:

Configurador Manual			
	Total	Total asignado	EXIBIR LAS ACTUAR
Laminado	Manual	340.380,00 €	0,00%
Troceado	Manual	198.098,00 €	0,00%
Tratamiento	Manual	278.690,00 €	0,00%
Auxiliar	Manual	112.340,00 €	0,00%
Administración	Manual	167.340,00 €	0,00%
Comercial	Manual	0,00 €	0,00%

Ilustración 44. Configurador sin asignar costes.

De primeras veríamos esta imagen. A la izquierda el total de las secciones, el tipo de imputación seleccionado con anterioridad, permitiendo cambiarlo con un desplegable, los costes totales de cada sección, y por último el porcentaje

asignado. Como se puede ver, vemos que tiene color amarillo. Esta página está configurada de tal manera que si no está repartido el total de los costes, sale en un color amarillo, si repartimos el 100%, se pondrá en verde, y si asignamos más del 100, se pondrá en rojo.

Asignamos el reparto correspondiente.

		Total	Total asignado	CUCHILLAS AFETAR
Laminado	Manual	340.680,00 €	100,00%	100%
Troceado	Manual	245.590,00 €	100,00%	100%
Tratamiento	Manual	278.400,00 €	100,00%	100%
Auxiliar	Manual	112.140,00 €	100,00%	100%
Administración	Manual	167.140,00 €	100,00%	100%
Comercial	Facturación	0,00 €	0,00%	

Ilustración 45. Configurador costes asignados.

Como podemos ver en la ilustración previa, se ha cambiado el tipo de imputación a la sección Comercial por facturación. Simplemente por mostrar al lector otra funcionalidad de formato, al seleccionar otra opción que no sea Manual, sale tachado, ya que entonces el configurador no afecta a esa sección.

Una vez tengamos terminado configurado el reparto por producto, podríamos dar al siguiente cuadro Imputación por productos, pero antes veamos el cuadro creado en este punto, la Hoja analítica por secciones:

	Laminado	Troceado	Tratamiento	Auxiliar	Administración	Comercial
Salidas y salario						
Trabajadores Laminado	117.500,00 €	-	-	-	-	-
Trabajador Troceado	-	117.500,00 €	-	-	-	-
Trabajador tratamiento	-	-	117.500,00 €	-	-	-
GASTOS						
1-Coste indirecto Laminado	203.180,00 €	-	-	-	-	-
2-Coste indirecto Troceado	-	108.090,00 €	-	-	-	-
3-Coste indirecto Tratamiento	-	-	140.900,00 €	-	-	-
4-Coste indirecto Auxiliar	-	-	-	112.140,00 €	-	-
5-Coste indirecto Administración	-	-	-	-	167.140,00 €	-
COSTE TOTAL	340.680,00 €	245.590,00 €	278.400,00 €	112.140,00 €	167.140,00 €	0,00 €

Ilustración 46. Imagen hoja Analítica por secciones.

Cuadro imputación por producto.

Ahora vamos a repartir los costes de las secciones en los diferentes productos, teniendo en cuenta el criterio elegido en el Configurador. Una vez realice el cálculo, automáticamente calculará el coste unitario por producto en la hoja Productos finales.

Lo primero que nos muestra la herramienta cuando realiza este cuadro es el siguiente mensaje:

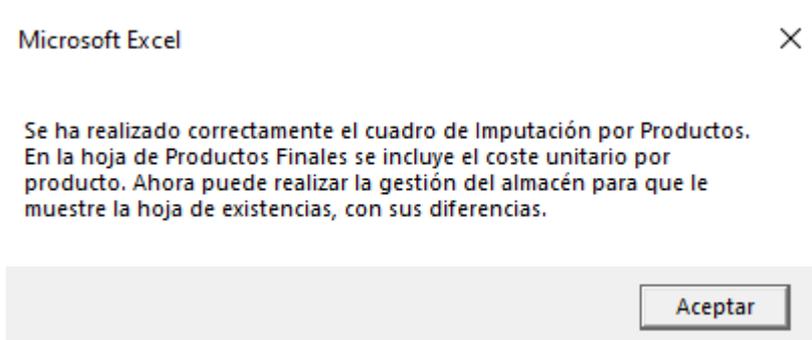


Ilustración 47. Mensaje de aviso.

Es un mensaje informativo sobre lo que acaba de calcular. Así como un aviso de que ahora si, la hoja de existencias estará completa al realizar la gestión del almacén . Antes de ver eso, mostremos el cuadro financiero calculado:

Cuchillas Afeitar		
Laminado	340.680,00 €	340.680,00 €
Troceado	245.590,00 €	245.590,00 €
Tratamiento	278.450,00 €	278.450,00 €
Auxiliar	112.140,00 €	112.140,00 €
Consumo Materias primas:		399.000,00 €
Productos en curso:	-	3.000,00 €
Productos Finales:	-	0,00 €
COSTE INDUSTRIAL		1.378.860,00 €

Ilustración 48. Imagen Cuadro Imputación por productos.

Si nos vamos a la hoja de productos finales, veremos lo que nos anunciaba herramienta con anterioridad:

Nombre	Existencias Iniciales €	PRODUCCION (Uds)	CONSUMIDO (Uds)	COSTE UNITARIO €	CONSUMO €	FINALES (Uds)
CUCHILLAS AFETAR	0	5.174.000	4.700.000	0,26 €	1.218.704,19 €	600.000

NOTA: Este valor calculado después de que se realice el Cuadro Inventario de secciones e productos.

Ilustración 49. Imagen Productos finales con coste unitario actualizado.

Como se puede apreciar, tendríamos un coste unitario por producto de 0,26€.

Gestión del almacén y hoja de existencias-diferencias.

Comenzamos haciendo la gestión del almacén. Aquí el programa cogerá las existencias iniciales, compras y salidas y calculará, mediante el método PMP, la gestión del almacén.

Para ello, pulsamos en el botón “Gestión del almacén”.

Una vez el programa realiza sus cálculos, nos muestra el siguiente mensaje por pantalla:

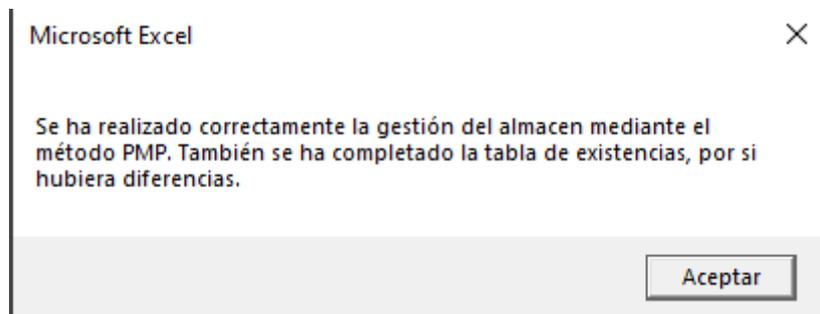


Ilustración 50. Mensaje de aviso.

Al pulsar el botón, como bien nos dice el anterior mensaje, no solo ha realizado la gestión del almacén, sino que ha completado la tabla de existencias con las posibles diferencias en las materias primas y productos. Es por este motivo por el cual la gestión del Almacén hay que realizarla después de haber generado el cuadro de imputación por productos.

Materia Prima	INICIAL	ENTRADAS	CONSUMOS	FINALES	AJUSTES
Pliego de acero	13.000,00 €	436.850,00 €	391.000,00 €	38.000,00 €	-2.850,00 €
CUCHILLAS AFETAR	0,00 €	1.378.860,00 €	1.218.704,19 €	153.947,90 €	6.157,92 €

Ilustración 51. Hoja existencia con ajustes calculados.

Como podemos ver, tenemos de ajustes negativos tanto en las materias primas como en los productos finales. Esto deberá tenerlo en cuenta en el cuadro final de Márgenes y Resultados.

Por último, veamos la tabla generada para la gestión del almacén.

Concepto	Materia prima	Fecha	Cantidad	Precio	Total	Cantidad	Almacén	Consumos totales
Inicial	Flejes de acero	01/12/2020	3200	9,30 €	29.760,00 €	3200	29,30 €	0,00 €
Compra	Flejes de acero	28/12/2020	43000	9,49 €	408.270,00 €	46000	9,30 €	0,00 €
consumo	Flejes de acero	30/12/2020	42000			4800	9,30 €	399.000,00 €

Ilustración 52. Imagen gestión del almacén (PMP)

Como se puede ver, ha organizado las existencia por fecha y ha calculado correctamente el sistema PMP. En el caso de que hubiera varias materias primas, las habría separado por una fila vacía.

Cuadro Márgenes y Resultados.

Y ya llegamos al final de esta guía de usuario, solo nos faltaría generar el cuadro de Márgenes y Resultados, para ver el resultado final del periodo, así como los distintos márgenes de la empresa.

Pulsamos en el botón Cuadro Márgenes y Resultados → Generar cuadro.

Cuenta de Márgenes y Resultados	
CUCHILLAS AFETAR	
Ventas	1.662.500,00 €
-Coste Industrial	-1.228.754,19 €
Margen Industrial	443.745,81 €
-Coste Comercial	0,00 €
Margen comercial	443.745,81 €
-coste administracion	-187.340,00 €
Rdo. de la Actividad	276.805,81 €
Ajustes	
Flejes de acero	-2.850,00 €
CUCHILLAS AFETAR	-4.157,92 €
Rdo. Periodo	267.597,90 €

Ilustración 53. Imagen cuadro de cuenta de Márgenes y Resultados.

Obtendríamos lo mostrado en la imagen anterior. Como nos muestra el programa, el coste comercial es cero, puesto que no le hemos asignado ningún gasto. También vemos como la herramienta tiene en cuenta los ajustes calculados en la hoja de existencias, obteniendo un Resultado de periodo de 267.597,90 €.

CAPÍTULO 4: ESTUDIO ECONÓMICO

Introducción.

En este capítulo se realizará un estudio económico del desarrollo de la herramienta de Contabilidad de Costes. Para ello, se ha tenido en cuenta tanto las horas invertidas en cada una de las fases que engloban el proyecto como los diferentes recursos utilizados para la consecución del mismo.

Para la realización del estudio económico, simularemos que no consiste en la realización de un trabajo de fin de grado, sino que se ha realizado un proyecto para un determinado cliente que ha solicitado a la empresa. Es por ello, que este proyecto constará de un director de proyecto, el cual controla y realiza el seguimiento del proyecto, y un Ingeniero en Organización industrial que ha realizado el desarrollo de la aplicación, así como de la documentación final.

Fases del proyecto.

Las fases del proyecto están condicionadas por el desarrollo de la herramienta, el cual es el principal objetivo de este proyecto. En la siguiente imagen podemos ver las siguientes etapas seguidas para la consecución de dicho objetivo.

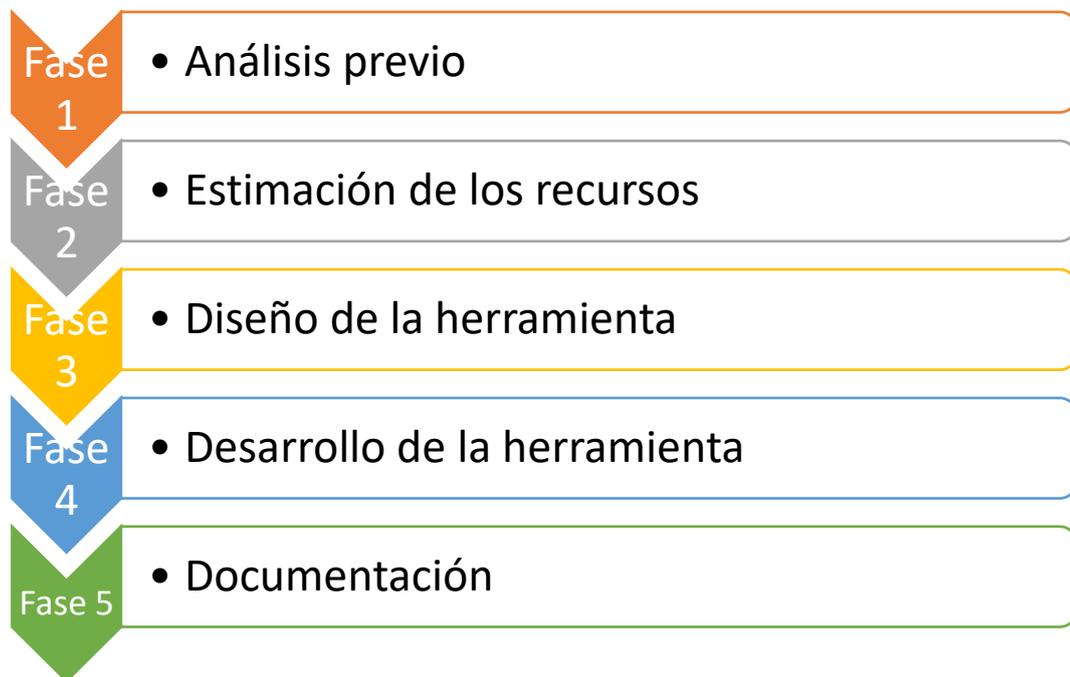


Ilustración 54: Fases del proyecto

A continuación, se explicará brevemente cada una de las etapas:

- **Fase 1. Análisis previo:** En esta etapa se analiza desde un punto de vista general el proyecto. Es en este punto donde se determina el alcance del mismo, así como afrontar los distintos retos que supone dicho proyecto. Para ello es importante realizar un minucioso análisis, así como buscar todo tipo de información para poder comprender al completo la complejidad del trabajo.
- **Fase 2. Estimación de los recursos.** Una vez realizado el análisis previo y teniendo clara la complejidad del mismo, así como los diferentes problemas que nos vamos a encontrar, realizamos un nuevo análisis para planificar en tiempo y recursos necesarios para desarrollar la herramienta. Es importante detectar la viabilidad del proyecto, ya que si el mismo resulta ser inviable, el detectarlo en sucesivas fases ocasionaría una pérdida de dinero.
- **Fase 3. Diseño de la herramienta:** Una vez confirmada la viabilidad del proyecto, tanto a nivel técnica como económica, hay que crear un diseño inicial de la herramienta. Con toda la información recabada en las fases anteriores, junto con las necesidades del cliente, se desarrollará un prototipo general de la aplicación, la cual puede ir sufriendo modificaciones una vez vaya avanzando el proyecto, pero el concepto general se mantendrá.
- **Fase 4. Desarrollo de la herramienta.** Cuando se tiene clara la línea de desarrollo a seguir, se comienza el desarrollo de la herramienta. Si bien es cierto que puede intercalarse la fase 3 y la 4 en algún punto del desarrollo de la herramienta, con el objetivo de solucionar y añadir nuevas funcionalidades a la herramienta. En esta etapa es importante testar la aplicación, una vez está terminada, con el objetivo de encontrar errores y depurarlos, garantizando el correcto funcionamiento de la misma.
- **Fase 5. Documentación.** En el momento que la herramienta ha sido desarrollada y cumple con los objetivos marcados, hay que desarrollar un manual de usuario para ayudar al usuario su utilización. También crear la correspondiente documentación que indique a nivel técnico como se ha desarrollado la misma, por futuras actualizaciones o por si en algún momento necesita depurar algún error.

Estudio económico.

Para realizar de la manera más precisa posible el estudio económico de este proyecto, se va a calcular los diferentes costes que han afectado al proyecto. Una vez tengamos estos costes calculados, se calculará el coste total del proyecto por fase, para tener este desglosado fase a fase.

Coste personal.

En lo que a personal se refiere, contamos con dos trabajadores: un director de proyecto y un ingeniero en Organización industrial. Para realizar el cálculo del coste de personal, se ha tenido en cuenta que el director de proyecto tiene un sueldo de 50.000 € y el Ingeniero en Organización Industrial tiene una retribución de 25000 €, a esto hay que sumarle el coste que supone la seguridad social a la empresa(28%).

En primer lugar, tendremos en cuenta los días laborales que se tendrían disponibles para trabajar. El resultado del cálculo se muestra en la siguiente tabla.

Días totales	365
Sábados y Domingos	-104
Días de vacaciones	-30
Días festivos	-14
Asuntos personales	-5
Días laborales reales 2020	212

Tabla 3: Días laborales disponibles 2020

Lo cual, si disponemos de un horario a jornada completa, obtendríamos un total de 1696 horas de trabajo disponibles anuales.

	Director del Proyecto	Ingeniero Organización industrial
Salario bruto/año	50.000,00 €	25.000,00 €
Seguridad social (28%)	14.000,00 €	7.000,00 €
Total €/año	64.000,00 €	32.000,00 €
Coste por hora (€/h)	37,74 €	18,87 €

Tabla 4: Coste trabajador por hora.

Como podemos ver en la tabla anterior, el coste por hora del director del proyecto es de 37,74 € y el del Ingeniero es de 18.87 €.

Coste material informático.

En este apartado debemos tener en cuenta el coste del ordenador, el cual fue comprado hace 7 años, por un importe de 775 €, viniendo consigo la licencia de Windows 8, que a su vez incluía la actualización a Windows 10. Para calcular el coste de este equipo, se ha tenido en cuenta el periodo de amortización linealmente de siete años, ya que es la antigüedad del mismo.

Para la realización del proyecto solo se ha necesitado el paquete Office, para los programas de Excel y Word.

En la siguiente tabla se muestra el coste total por hora que supone el equipo informático para la realización del proyecto.

Concepto	Coste	Unidades	Amortización	Coste por hora
HP ENVY 17-j020ss Intel Core i7-4700MQ (4x2,40 GHz / 6 MB caché) + Licencia Windows 10.	775,00 €	1	7	0,07
Licencia Microsoft Office	135,00 €	1	1	0,08
Coste total por hora (€/h)				0,14 €

Tabla 5: Coste materiales informáticos

Costes indirectos.

En este punto recogemos todos los gastos externos que no tienen que ver directamente con el desarrollo del proyecto, pero que suponen un coste para la óptima realización del mismo. Estos costes, en algunos casos, pueden ser más difíciles de calcular, por lo que se ha tenido una estimación aproximada de lo que podría ser una situación real.

Concepto	Coste
Electricidad	190,00 €
Internet	233,70 €
Alquiler	2.400,00 €
Otros	100,00 €
Total	2.923,70 €
Coste por hora (€/h)	1,72 €

Tabla 6. Costes indirectos

Coste económico del proyecto.

En este punto se tendrá en cuenta los costes calculados con anterioridad, diferenciando en cada fase del proyecto, para así tener los costes desglosados uno a uno y ver con mayor claridad el reparto de los mismos.

Coste fase 1. Análisis previo.

En esta fase se tiene en cuenta que el director del proyecto ha realizado varias reuniones con el cliente hasta comprender las exigencias de los mismos. El Ingeniero deberá dar soporte al director del proyecto.

Fase 1: ANÁLISIS PREVIO			
	Coste unitario(€/h)	Horas	Coste total
Coste Personal/Director Proyecto	37,74 €	40	1.509,43 €
Coste Personal/Ingeniero Org. Industrial.	18,87 €	10	188,68 €
Coste equipo informático	0,14 €	10	1,45 €
Costes indirectos	1,72 €	10	17,24 €
COSTE TOTAL			1.716,80 €

Tabla 7: Coste fase 1.

Coste fase 2: Estimación de recursos.

En esta fase, el ingeniero deberá calcular la duración de cada tarea para calcular el tiempo y los recursos necesarios para la realización del proyecto. Quedaran definidos los requisitos necesarios para cumplir los objetivos y el alcance fijados con anterioridad. En este punto, el director del proyecto supervisará las conclusiones del ingeniero, por si hubiera algún error o punto de mejora.

Fase 2: ESTIMACIÓN DE LOS RECURSOS			
	Coste unitario(€/h)	Horas	Coste total
Coste Personal/Director Proyecto	37,74 €	20	754,72 €
Coste Personal/Ingeniero Org. Industrial.	18,87 €	40	754,72 €
Coste equipo informático	0,14 €	40	5,80 €
Costes indirectos	1,72 €	40	68,96 €
COSTE TOTAL			1.584,18 €

Tabla 8: Coste fase 2

Coste fase 3. Diseño de la herramienta.

Periodo el cual se recopila toda información posible referente al proyecto. Búsqueda de proyectos similares a nivel interno como externo, teniendo presentes las exigencias y necesidades del cliente. La función del director del proyecto es servir de apoyo al ingeniero y supervisar que se esté cumpliendo con los objetivos acordados.

Fase 3: DISEÑO DE LA HERRAMIENTA			
	Coste unitario(€/h)	Horas	Coste total
Coste Personal/Director Proyecto	37,74 €	8	301,89 €
Coste Personal/Ingeniero Org. Industrial.	18,87 €	75	1.415,09 €
Coste equipo informático	0,14 €	75	10,87 €
Costes indirectos	1,72 €	75	129,29 €
COSTE TOTAL			1.857,14 €

Tabla 9:Coste fase 3

Coste fase 4. Desarrollo de la herramienta.

Esta es la fase más compleja del proyecto. Una vez se tienen definidos los objetivos y se tiene documentado el diseño de la herramienta, se comienza con el desarrollo de la herramienta. Es la fase más larga con diferencia y la que determina el resultado del proyecto.

Fase 4: DESARROLLO DE LA HERRAMIENTA			
	Coste unitario(€/h)	Horas	Coste total
Coste Personal/Director Proyecto	37,74 €	8	301,89 €
Coste Personal/Ingeniero Org. Industrial.	18,87 €	215	4.056,60 €
Coste equipo informático	0,14 €	215	31,15 €
Costes indirectos	1,72 €	215	370,63 €
COSTE TOTAL			4.760,27 €

Tabla 10: Coste fase 4

Coste fase 5. Documentación.

Constituye la creación del documento final de proyecto, que recoge la guía del programador y la del usuario. Así como toda documentación técnica que sirva de utilidad. Es tarea concreta del ingeniero, que presenta en una reunión final al director del proyecto para que este lo valide.

Fase 5: DOCUMENTACIÓN			
	Coste unitario(€/h)	Horas	Coste total
Coste Personal/Director Proyecto	37,74 €	4	150,94 €
Coste Personal/Ingeniero Org. Industrial.	18,87 €	45	849,06 €
Coste equipo informático	0,14 €	45	6,52 €
Costes indirectos	1,72 €	45	77,57 €
COSTE TOTAL			1.084,09 €

Tabla 11: Coste fase 5

Coste total del proyecto.

Teniendo en cuenta los costes calculados con anterioridad, nos da un coste total de 11.002,49€, como podemos ver en la siguiente tabla.

FASE	COSTE
Fase 1: Análisis previo	1.716,80 €
Fase 2: Estimación de los recursos	1.584,18 €
Fase 3: Diseño de la herramienta	1.857,14 €
Fase 4: Desarrollo de la herramienta	4.760,27 €
Fase 5: Documentación	1.084,09 €
COSTE TOTAL	11.002,49 €

Tabla 12: Coste total del proyecto

Si vemos los costes en porcentajes, vemos cómo la fase del desarrollo de la herramienta se lleva un 43% del total, siendo esta la fase con más costes con diferencia. Como ya se había comentado antes, esta fase es la más compleja e importante para la consecución de los objetivos.



Ilustración 55: Gráfico costes totales por fases.

Precio de venta del producto.

Una vez hemos calculado el coste de producción de la herramienta, pasamos a calcular el precio de venta del producto, así como la factura final.

Para calcular el precio de venta nos valemos de la siguiente fórmula:

$$\text{Precio de venta} = \frac{\text{Coste producto}}{1 - \text{Margen beneficio}}$$

El director del proyecto ha calculado que en este proyecto hay que obtener un 20% de beneficio, por lo que, aplicando la fórmula anterior, obtenemos un **precio de venta de 13.753,11 €**. Teniendo en cuenta que el IVA es del 21%, la factura alcanza un total de 16.641.26 €

CONCLUSIONES Y LÍNEAS FUTURAS DE DESARROLLO

Introducción.

Es momento de hacer análisis del trabajo realizado durante mucho tiempo hasta llegar hasta aquí. En este punto se darán las conclusiones obtenidas una vez realizado el presente trabajo fin de grado, así como ideas de unas potenciales líneas de desarrollo.

Conclusiones.

Después de muchas horas dedicadas para la consecución de este trabajo, podemos concluir que no solo se ha cumplido con el principal objetivo del proyecto, el cual era crear una herramienta en Excel para la contabilidad de costes, sino que, además, esta incluye las siguientes características que se detallan a continuación:

- La herramienta ha sido programada en su totalidad con el complemento de desarrollador. Implementando, no solo los formularios, si no todas las macros necesarias para realizar todos los cálculos. Permitiendo con ello uno de los objetivos principales: que la aplicación fuera lo más flexible posible, sin verse limitada por ella misma. Véase en número de secciones, productos, costes indirectos... y un largo etcétera.
- En línea con la mencionada flexibilidad del programa que comentábamos antes, la herramienta dispone de una serie de funcionalidades que permiten, o bien utilizar unos tipos de imputación que la propia aplicación elige por ti, o que configures en porcentaje el reparto de los costes como el usuario considere oportuno, dando total control de los mismo y aumentando con ello la mencionada flexibilidad.
- Una herramienta que no solo se ha trabajado en su apariencia a nivel estético, sino que se ha trabajado en ella a nivel de funcionalidad, editando la cinta de opciones para que sea más profesional y mucho más intuitiva para el usuario.

Y tan importante es su aspecto visual así como su funcionamiento, si no toda la información que aporta para la toma de decisiones:

- Gestión del almacén mediante el método PMP, cálculo imputación por secciones y por producto, coste unitario por producto y por último, cuenta de márgenes y resultados, teniendo en cuenta los ajustes de existencia que ocurren en el periodo contable.

Es por ello por lo que a título personal me encuentro muy satisfecho con el resultado final del programa, ya que ha superado mis expectativas. No solo se

han ampliado mis conocimientos sobre la contabilidad de costes, si no que he adquirido muchos conocimientos en Excel, en la automatización de tareas mediante macros, así como en la ejecución de un proyecto, cumpliendo objetivos y tiempos.

Líneas futuras de desarrollo.

En esta sección del presente trabajo me gustaría indicar, que para el autor del mismo, no es más que una herramienta en fase inicial de desarrollo, quedando una grandísima cantidad de opciones para aumentar las posibilidades de la herramienta, las cuales por cuestiones de tiempo no han sido posible implementarlas.

Si bien, la herramienta realiza los diferentes cálculos de la contabilidad de costes, las numerosas mejoras que se podrían implementarse son casi infinitas. Por destacar algunas:

- Añadir más alternativas en la gestión del almacén, permitiendo al usuario determinar por materia prima, qué tipo de método quiere que realice el cálculo (LIFO,FIFO... etcétera)
- Crear formularios para la modificación de los datos añadidos por el usuario, haciéndolo mucho más intuitivo y funcional.
- Generar un login con contraseña que, según el usuario que entre, active o desactive funcionalidades de la cinta de opciones, como por ejemplo:

Si quien inicia sesión es la encargada/o de recursos humanos, solo tenga activa en la cinta de opciones el botón de añadir, modificar y eliminar personal. En cambio, si quien inicia sesión es administrador/a, todos los botones estén activos. Con esto permitiríamos un mayor control de la herramienta.

- Añadir la posibilidad de incluir de una manera fácil y sencilla políticas de empresa que afecten a los costes, y estos sean tenidos en cuenta en el cálculo final.
- Una última funcionalidad que según el margen de beneficio que se quiera obtener por producto, la propia herramienta calcule el precio de venta óptimo, así ver de una manera intuitiva la rentabilidad por producto.

BIBLIOGRAFÍA

Bibliografía

- Amelot, M. (2016). *VBA EXCEL 2016: Programación en Excel: Macros y lenguaje VBA*. Ediciones ENI.
- Castro, A. G. (Octubre de 2018). Trabajo Fin de Grado. GENERADOR DE EJERCICIOS PARA ANÁLISIS DE ESTADOS CONTABLES. Valladolid, España.
- Fraile, A. V. (Noviembre de 2018). Trabajo Fin de Grado .GESTIÓN DE PERSONAL A TRAVÉS DEL MÉTODO DE LOS PERFILES. Valladolid, España.
- Fullana Belda, C., & Ortega Paredes, J. L. (2008). *Manual Contabilidad de Coste*. Delta.
- González, A. E. (Septiembre de 2015). Trabajo Fin de Grado. Desarrollo de una herramienta para el cálculo de costes a partir del diagrama general de imputación de costes y de las cuentas anuales de la empresa. Valladolid, España.
- Mallo Rodríguez, C., & Jiménez Montañés, M. Á. (2009). *Contabilidad de Costes*. Madrid: Ediciones Pirámide.
- Sáez Torrecilla, Á., Fernández Fernández, A., & Gutiérrez Díaz, G. (2009). *Contabilidad de Costes y Contabilidad de Gestión*. McGraw-Hill/Interamericana de España, S.A.U.
- Schneider, E. (1949). *Contabilidad Industrial*. Aguilar.
- Seco, J. A. (1999). *Introducción a la Contabilidad de Gestión*. UNIVERSIDAD PONTIFICIA DE COMILLAS.

Otras fuentes utilizadas

Bert toolkit. (2018). <https://bert-toolkit.com>. Obtenido de <https://bert-toolkit.com>: <https://bert-toolkit.com/imagenes-so-list.html>

Celemín, S. A. (2019). <https://ayudaexcel.com/>. Obtenido de <https://ayudaexcel.com/la-funci%C3%B3n-msgbox-de-vba-a-fondo/>

Muñoz, M. (2019). <https://tutorialexcel.com>. Obtenido de <https://tutorialexcel.com:/macro-para-abrir-la-calculadora/>

Programacion office VBA. (2019). <https://programacionofficevba.blogspot.com>. Obtenido de <https://programacionofficevba.blogspot.com/p/pagina-seis.html>

Wiki Openoffice. (2013). <https://wiki.openoffice.org/>. Obtenido de <https://wiki.openoffice.org/wiki/ES/Manuales/GuiaA00/TemasAvanzados/Macros/StarBasic/TrabajandoConFormularios>

Ayuda Excel (2020): <https://ayudaexcel.com>. Obtenido de <https://ayudaexcel.com/foro/forum/6-macros-y-programaci%C3%B3n-vba/>

ANEXOS

Anexos

Código Cinta de opciones.

```

<customUI xmlns="http://schemas.microsoft.com/office/2009/07/customui">
  <ribbon startFromScratch="false">
    <tabs>
      <tab id="customTab" label="Contabilidad de Costes">
        <group id="customGroup">
          <button id="customButton" label="Inicio"
imageMso="BlogHomePage" size="large" screentip="Menú principal"
onAction="menu"/>
        </group>
        <group id="customGroup1" label="Información
general">
          <menu id="MyMenu2" label="Datos
generales" imageMso="CategorizeMenu" screentip="Datos generales
descripción general de empresa" supertip="Podemos añadir
la información general
description label empresa,desde la dirección, como la estructura de la
misma" size="large">
            <button id="customButton1"
label="Informacion " imageMso="PropertySheet" screentip="Datos generales"
supertip="Nombre de la entidad,dirección."
onAction="lanzar_fm_datosempresa"/>
            <button id="customButton2"
label="Estructura" imageMso="ProjectManageDeliverables"
screentip="Estructura de la empresa"
supertip="Añadimos las
distintas secciones, asi como se asignan los costes en ellas."
onAction="lanzar_fm_estructura"/>
            <button id="customButton211"
label="Consultar Datos Generales" imageMso="DatabaseDocumenter"
screentip="Acceder base dedatos" onAction="verempresa"/>
          </menu>
        </group>
        <group id="customGroup2" label="Gestión de
existencias">
          <menu id="MyMenu211" label="Productos
finales y productos en curso" imageMso="BlogOpenExisting" size="large">
            <button id="customButton4"
label="Añadir Productos Finales y en curso." imageMso="BlogOpenExisting"
screentip="Productos finales y en curso de la empresa"
supertip="Nos permite añadir
los productos finales como los que tenemos en curso."
onAction="lanzar_fm_productos"/>
            <button id="customButton411"
label="Consultar los Productos Finales" screentip="Productos Finales"
supertip="Podemos ver los
Productos finales de la compañía, asi como el coste unitario por producto."
imageMso="DatabaseDocumenter" onAction="verpf"/>
            <button id="customButton421"
label="Consultar los Productos en Curso " imageMso="DatabaseDocumenter"
onAction="verPC"/>
          </menu>
          <menu id="MyMenu543" label="Existencias
Iniciales" imageMso="ProjectManageDeliverables" size="large">

```

```

label="Existencias"
screentip="Existencias iniciales"
<button id="customButton512"
imageMso="ProjectManageDeliverables"
supertip="En esta
sección, recogemos las existencias iniciales de las materias primas."
onAction="lanzar_fm_existencias"/>
<button id="customButton1422"
label="Consultar "
imageMso="DatabaseDocumenter"
onAction="verexistencias"/>
</menu>
</group>
<group id="customGroup222" label="Gestión del
almacen">
<menu id="MyMenu333" label="Compras de
materias primas" imageMso="MailMergeGotToLastRecord" size="large">
<button id="customButton331"
label="Compras de materias primas" imageMso="MailMergeGotToLastRecord"
screentip="Compras materias primas" onAction="lanzarcompra"/>
<button id="customButton413"
label="Consultar base de datos"
imageMso="DatabaseDocumenter"
onAction="comprasmp"/>
</menu>
<menu id="MyMenu66" label="Consumo de
materias primas" imageMso="MailMergeGoToFirstRecord" size="large">
<button id="customButton66"
label="Consumo de materias primas " imageMso="MailMergeGoToFirstRecord"
onAction="venta"/>
<button id="customButton56"
label="Consultar base de datos"
imageMso="DatabaseDocumenter"
onAction="verconsumo"/>
</menu>
<menu id="MyMenu3332" label="Gestión del
almacen" imageMso="FileCheckOutDiscard" size="large">
<button id="customButton4233"
label=" Realizacion de método PMP " imageMso="FileCheckOutDiscard"
onAction="almacen"/>
</menu>
<button id="customButton57"
label="Consultar base de datos"
imageMso="DatabaseDocumenter"
onAction="veralmacen"/>
</menu>
<button id="customButton5347"
label="Consultar Cuadro de Existencias finales"
imageMso="DatabaseDocumenter" size="large"
screentip="Hoja existencias
finales." supertip="Podemos ver el cuadro final de entradas, salidas y
ajustes por cada materia prima." onAction="existenciasfinales"/>
</group>
<group id="customGroup3" label="Recursos
Humanos">
<menu id="MyMenu3" label="Personal"
imageMso="AddOrRemoveAttendees" screentip="Recursos Humanos"
supertip="Añadir toda la
información correspondiente al personal de la empresa: salario, fecha de
inicio, fecha de despido, así como en qué sección trabaja" size="large">
<button id="customButton6"
label="Añadir Empleado"
imageMso="DistributionListAddNewMember"
onAction="lanzar_fm_personal"/>
</menu>
<button id="customButton7"
label="Consultar base de datos del personal" imageMso="AccessListContacts"
size="large" onAction="bdpersonal"/>
</group>

```

```

        <group id="customGroup4" label="Finanzas">
            <menu id="MyMenu4" label="Facturación"
imageMso="PropertySheet" screentip="Facturación por producto."
                supertip="Añadir la
facturación obtenida por los diferentes productos o servicios que ofrece
la empresa" size="large">
                <button id="customButton8"
label="Facturación"
                imageMso="AccountingFormat"
onAction="lanzar_fm_facturacion"/>
                <button id="customButton81"
label="Base de datos de Facturación" imageMso="DatabaseDocumenter"
onAction="bdfacturacion"/>
            </menu>
            <menu id="MyMenu5" label="Gastos"
imageMso="ChartTypeOtherInsertGallery" screentip="Gastos generales"
                supertip="Añadir toda la
información correspondiente al gasto en la empresa: tipo de
gasto,cantidad,fechas en las que ocurre, asi como a cual o cuales secciones
se les imputan."
                size="large">
                <button id="customButton9"
label="Gastos"
                imageMso="ChartTypeOtherInsertGallery"
onAction="lanzar_fm_gastos"/>
                <button id="customButton91"
label="Base de datos de Gastos" imageMso="DatabaseDocumenter"
onAction="bdgastos"/>
            </menu>
        </group>
        <group id="customGroup5" label="Informes
Financieros ">
            <menu id="MyMenu6" label="Hoja Analitica
por secciones" imageMso="ChartEditDataSource" size="large">
                <button id="customButton10"
label="Generar cuadro" imageMso="DatabaseDocumenter" onAction="coste"/>
                <button id="customButton101"
label=" Ver hoja Analitica por Secciones" imageMso="DatashheetColumnLookup"
onAction="bdcoste"/>
            </menu>
            <menu id="MyMenu7" label="Cuadro
Imputación de secciones a productos" imageMso="CustomGallery1"
size="large">
                <button id="customButton11"
label="Generar cuadro" imageMso="ChartAreaChart" onAction="cuadropf"/>
                <button id="customButton111"
label=" Ver cuadro de Imputación por Productos"
imageMso="DatashheetColumnLookup" onAction="bdcuadropf"/>
            </menu>
            <menu id="MyMenu8" label="Cuadro Márgenes
y Resultados" imageMso="ControlChart" size="large">
                <button id="customButton13"
label="Generar cuadro" imageMso="ChartInsert" onAction="margenes"/>
                <button id="customButton131"
label="Ver Margenes y Resultados" imageMso="DatashheetColumnLookup"
onAction="bdmargenes"/>
            </menu>
        </group>
        <group id="customGroup6" label="Configuración">
            <button id="customButton14" label="Tipos
de imputación" imageMso="PageMenu" screentip=" Configurador de los tipos de
imputación"
                supertip=" Nos permite asignar con
total flexibilidad las diferentes imputaciones de las secciones a
productos." size="large" onAction="configurador"/>
        </group>
        <group id="customGroup62" label="RESET">
    
```

```

                                <button id="customButton22"
label="Limpiar todo " imageMso="RefreshMenu" screentip=" Eliminar todos los
datos" supertip=" Elimina todos los registros introducidos en la
aplicación "
                                size="large" onAction="borrar"/>
                                </group>
                                </tab>
                                </tabs>
                                </ribbon>
</customUI>
```

Código Visual Basic.

Módulos.

Código Módulo A1_Hoja_Analitica_secciones.

```
Sub coste(registro As IRibbonControl)
Application.ScreenUpdating = False

    Dim c As Range, d As Range, columna As Long, fila, celda

    'arrastramos columna 1 de la hoja de personal y gastos

    Hoja6.Activate
    Range("A2") = "=CONCATENATE(B2,E2)"
    With Range("a2")
        .AutoFill .Resize(.Offset(, 1).End(xlDown).Row - .Row + 1),
xlFillDefault
    End With
    Hoja7.Activate
    Range("A2") = "=CONCATENATE(B2,E2)"
    With Range("a2")
        .AutoFill .Resize(.Offset(, 1).End(xlDown).Row - .Row + 1),
xlFillDefault
    End With

    Hoja11.Activate

    Cells.ClearContents ' limpiamos la hoja primero

    Range("D1") = " Cuadro Imputacion por secciones"

    ' primero hacemos la parte horizontal de arriba (centros de
coste)

    Hoja2.Activate

    Range("D2", Range("D2").End(xlDown)).Select
    Selection.Copy

    Hoja11.Activate
    Range("C3").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:=
False, Transpose:=True
    Application.CutCopyMode = False
```

```

'traemos los gastos (columna izquierda)

Hoja6.Activate

Range("B2", Range("B2").End(xlDown)).Select
Selection.Copy

Hoja11.Activate
Range("B4") = "Sueldos y Salarios"
Range("B5").PasteSpecial xlPasteAll 'copia lo seleccionado en
la hoja6

Hoja7.Activate

Range("B2", Range("B2").End(xlDown)).Select
Selection.Copy

Hoja11.Activate

Cells(Rows.Count, 2).End(xlUp).Offset(1, 0).Select
ActiveCell.Offset(0, 0) = "GASTOS"

ActiveCell.Offset(1, 0).PasteSpecial xlPasteAll

Application.CutCopyMode = False

' buscamos los gastos en cada hoja

Hoja11.Activate

With Range("c4")
    .Formula
    "=iferror(vlookup(concatenate($b4,c$3),personal!$a:$i,9,false),iferror(vlookup(concatenate($b4,c$3),gastos!$a:$i,9,false),"-"))"
    .AutoFill .Resize(, .Offset(-1).End(xlToRight).Column
.Column + 1), xlFillDefault
    With .Resize(, .Offset(-1).End(xlToRight).Column
.Column + 1)
        .AutoFill .Resize(.Offset(, -1).End(xlDown).Row
.Row + 1), xlFillDefault
    End With
End With

' calculamos los costes totales

Cells(Rows.Count, 2).End(xlUp).Offset(1, 0).Select
ActiveCell.Offset(0, 0) = "COSTE TOTAL"

With ActiveCell.Offset(, 1)

    With Hoja11

```

```
        fila = .Range("C" & Rows.Count).End(xlUp).Row '
Última Fila ocupada
        columna = .Cells(3,
Cells.Columns.Count).End(xlToLeft).Column ' Última Columna ocupada
        Set d = .Range(.Cells(4, 3), .Cells(4, columna)) 'Creo
un Rango de la primera fila
        For Each celda In d ' Paso por todas las celdas del
rango de la primera fila
            Set c = .Range(Cells(3, celda.Column),
Cells(fila, celda.Column)) ' creo un rango con la columna a sumar
            .Cells(fila + 1, celda.Column) =
Application.WorksheetFunction.Sum(c) ' sumo el rango con la funcion
Sum
            Set c = Nothing ' vacio por si acaso
        Next celda
    End With

End With

'-----

'CONFIGURADOR

'-----
Hoja16.Activate
Cells.Select
    Selection.ClearContents

Range("A3") = 1
Range("B3") = 2

Hoja3.Activate
If IsEmpty(ActiveSheet.Cells(3, 2)) Then
    Range("B2").Select
    Selection.Copy

Hoja16.Activate
Range("E3").PasteSpecial Paste:=xlValues,
Operation:=xlNone, SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

Else
```

```

Hoja3.Activate
Range("B2", Range("B2").End(xlDown)).Select
Selection.Copy

Hoja16.Activate
Range("E3").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

End If

Hoja11.Activate

Dim uF As Long, uC As Long

' calculamos cual es la última fila y columna de la tabla
para copiar los costes por sección

uF = Range("C" & Rows.Count).End(xlUp).Row
uC = Cells(uF, Columns.Count).End(xlToLeft).Column

Range(Cells(uF, 2), Cells(uF, uC)).Offset(, 1).Copy '
copiamos los costes

Hoja16.Activate
' pegamos los costes haciendo la transpuesta
Range("C4").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

Hoja16.Activate

Range("A4") = "'DATOS EMPRESA'!D2"

With Range("A4")

.AutoFill .Resize(, .Offset(2).End(xlToRight).Column -
.Column), xlFillDefault
With .Resize(, .Offset(1).End(xlToRight).Column -
.Column)
.AutoFill .Resize(.Offset(, 2).End(xlDown).Row - .Row
+ 1), xlFillDefault
End With
End With
Range("A2") = 1

Range("B2") = 2
Range("C3") = "Total"
Range("D3") = "Total asignado"
Range("D4") = "=SUM(E4:AAA4)"
Range("A2:b2").Select
With Selection

```

```

        .AutoFill .Resize(, .Offset(1, 1).End(xlToRight).Column
- .Column + 1), xlFillDefault

        End With

        With Range("D4")

            .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row
+ 1), xlFillDefault

            End With

            Range("E1") = "Configurador Manual"
' -----
'           '   FORMATO
' -----

        Range("A3:B3").Select

        With Selection.Font
            .ThemeColor = xlThemeColorDark2
            .TintAndShade = -0.499984740745262
        End With

        Rows("2:2").Select
        With Selection.Font
            .ThemeColor = xlThemeColorDark2
            .TintAndShade = -0.499984740745262
        End With

        Columns("A:L").Select
        Selection.ColumnWidth = 21.14

        'inserta despegable
        Range("B4", Range("B4").End(xlDown)).Select

        With Selection.Validation
            .Delete
            .Add                                     Type:=xlValidateList,
AlertStyle:=xlValidAlertStop, Operator:= _
            xlBetween,
Formula1:="=IF(OR(A4=""Administracion"",A4=""Comercial""),Menú!$B$
1:$B$3,Menú!$A$1:$A$2)"
            .IgnoreBlank = True
            .InCellDropdown = True
            .InputTitle = ""
            .ErrorTitle = ""
            .InputMessage = ""
            .ErrorMessage = ""
            .ShowInput = True
            .ShowError = True
        End With

        ' Hoja1.Activate

End Sub

```

Código módulo A2_Imputacion_Productos

```

Sub cuadropf(registro As IRibbonControl)

    Application.ScreenUpdating = False
    Dim c As Range, d As Range, columna As Long, fila, celda

    Hoja12.Activate
    Cells.ClearContents

    Range("d1") = " Imputacion por productos"

    Hoja3.Activate
    If IsEmpty(ActiveSheet.Cells(3, 2)) Then
        Range("B2").Select
        Selection.Copy
        Hoja12.Activate
        Range("D3").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:= _
        False, Transpose:=True
        Application.CutCopyMode = False

    Else
        Hoja3.Activate
        Range("B2", Range("B2").End(xlDown)).Select
        Selection.Copy

        Hoja12.Activate

        Range("D3").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:= _
        False, Transpose:=True
        Application.CutCopyMode = False

    End If

    Hoja11.Activate
    Range("C3", Range("C3").End(xlToRight)).Select
    Selection.Copy

    Hoja12.Activate
    Range("B4").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:= _
    False, Transpose:=True
    Application.CutCopyMode = False

    'calculamos el porcentaje facturado por producto
    Hoja10.Activate

    If ActiveSheet.Cells(1, 5) = 0 Then

        Hoja10.Activate

        MsgBox "No ha introducido facturacion. Introduce la
facturación y vuelve a ejecutar el programa"

        Exit Sub

    End If
    Hoja10.Activate

```

```

Range("B2:C2", Range("B2").End(xlDown)).Select
Selection.Copy

Hoja8.Activate

Range("b2").PasteSpecial Paste:=xlValues
Application.CutCopyMode = False

Hoja10.Activate

Range("B2:C2", Range("B2").End(xlDown)).Select
Selection.Copy
'-----
'FACTURACION AUX
'-----
Hoja8.Activate

With Range("e1")
    .Formula = "=Sum(C:C)"
End With

    If IsEmpty(ActiveSheet.Cells(3, 2)) Then
        Range("D2").Select
        Selection.Copy
        Hoja12.Activate
        Range("D2").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:=False, Transpose:=True
        Application.CutCopyMode = False

    Else

        With Range("D2")
            .Formula = "=(C2/$e$1)"

            With .Resize(, .Offset(-1).End(xlToRight).Column
.Column + 1)
                .AutoFill .Resize(.Offset(, -1).End(xlDown).Row
.Row + 1), xlFillDefault
            End With
        End With

        Range("D2", Range("d2").End(xlDown)).Select
        Selection.Copy

        Hoja12.Activate

        Range("D2").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:=
        False, Transpose:=True
        Application.CutCopyMode = False
    End If

'-----
Hoja11.Activate

Dim uF As Long, uC As Long

```

```

        ' calculamos cual es la última fila y columna de la tabla para
        copiar los costes por sección

        uF = Range("C" & Rows.Count).End(xlUp).Row
        uC = Cells(uF, Columns.Count).End(xlToLeft).Column

        Range(Cells(uF, 2), Cells(uF, uC)).Offset(, 1).Copy

' copiamos los costes

        Hoja12.Activate
        ' pegamos los costes haciendo la transpuesta
        Range("C4").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:=
        False, Transpose:=True
        Application.CutCopyMode = False

        With Range("A4")

            .Formula = "=vlookup($B4,'Configurador'!$A:$B,2,false)"
            .AutoFill .Resize(.Offset(, 1).End(xlDown).Row - .Row + 1),
xlFillDefault

            End With

        If IsEmpty(ActiveSheet.Cells(3, 5)) Then

            With Range("D4")

                .Formula =
                "=IF($A4=""Facturacion"", $C4*D$2, $C4*vlookup($B4, 'Configurador'!$1:$10485
276, 'Configurador'!E$2, false))"

                .AutoFill .Resize(.Offset(, -1).End(xlDown).Row -
                .Row + 1), xlFillDefault

            End With

        Else

            With Range("D4")

                .Formula =
                "=IF($A4=""Facturacion"", $C4*D$2, $C4*vlookup($B4, 'Configurador'!$1:$10485
276, 'Configurador'!E$2, false))"

                .AutoFill .Resize(, .Offset(-1).End(xlToRight).Column
                - .Column + 1), xlFillDefault
                With .Resize(, .Offset(-1).End(xlToRight).Column -
                .Column + 1)

                    .AutoFill .Resize(.Offset(, -1).End(xlDown).Row -
                    .Row + 1), xlFillDefault
                End With
            End With

        End If

        Range("b4").Select ' eliminamos la sección Administracion

        Do While Not IsEmpty(ActiveCell)
            If ActiveCell.Value = "Administracion" Then
                Selection.EntireRow.Delete
            Else
                ActiveCell.Offset(1, 0).Select
            End If
        End While
    
```

```

        Loop
        Range("b4").Select
        Do While Not IsEmpty(ActiveCell) 'Eliminamos la sección
Comercial
            If ActiveCell.Value = "Comercial" Then
                Selection.EntireRow.Delete
            Else
                ActiveCell.Offset(1, 0).Select
            End If
        Loop
        Range("c4").Select ' eliminamos las secciones sin gastos
        Do While Not IsEmpty(ActiveCell)
            If ActiveCell.Value = "0" Then
                Selection.EntireRow.Delete
            Else
                ActiveCell.Offset(1, 0).Select
            End If
        Loop

        Cells(Rows.Count, 2).End(xlUp).Offset(1, 0).Select
        ActiveCell.Offset(0, 0) = "Consumo Materias primas:"

        Hoja19.Activate

        If Not IsEmpty(ActiveSheet.Cells(1, 4)) Then
' hay dos productos o mas
            If Not IsEmpty(ActiveSheet.Cells(3, 1)) Then
' hay dos o mas mp y 2 p

                Range("C2", Range("C2").End(xlDown).End(xlToRight)).Copy

                Hoja12.Activate
                ActiveCell.Offset(, 1) = "-"
                ActiveCell.Offset(, 2).PasteSpecial xlPasteValues
                Cells(Rows.Count, 4).End(xlUp).Offset(1, 0).Select
                ActiveCell.Offset(0, -2) = "Productos en curso:"
                ActiveCell.Offset(0, -1) = "-"

            Else ' 2 prod y 1 mp

                Range("C2", Range("C2").End(xlToRight)).Copy

                Hoja12.Activate
                ActiveCell.Offset(, 1) = "-"
                ActiveCell.Offset(, 2).PasteSpecial
xlPasteValues

                Cells(Rows.Count, 4).End(xlUp).Offset(1, 0).Select
                ActiveCell.Offset(0, -2) = "Productos en curso:"
                ActiveCell.Offset(0, -1) = "-"
                End If

            Else 'solo 1 producto
' hay dos o mas mp y 2 p

```

```

        Range("C2", Range("C2").End(xlDown)).Copy

        Hoja12.Activate

        ActiveCell.Offset(, 2).PasteSpecial xlPasteValues
        Cells(Rows.Count, 4).End(xlUp).Offset(1, 0).Select
        ActiveCell.Offset(0, -2) = "Productos en curso:"
        ActiveCell.Offset(0, -1) = "-"

        Else ' 1 producto 1mp
            Range("C2").Copy
            Hoja12.Activate

            ActiveCell.Offset(, 2).PasteSpecial
xlPasteValues
            Cells(Rows.Count, 4).End(xlUp).Offset(1, 0).Select
            ActiveCell.Offset(0, -2) = "Productos en curso:"
            ActiveCell.Offset(0, -1) = "-"

            End If

        End If

        Hoja4.Activate
        If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then
            Range("C2",
Range("C2").End(xlDown)).Select
            Selection.Copy

            Hoja12.Activate
            ActiveCell.Offset(, 0).PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
            False, Transpose:=True
            Application.CutCopyMode = False

            Else

            Range("C2").Select
            Selection.Copy
            Hoja12.Activate
            'se corrigio esto
            ActiveCell.Offset(, 0).PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
            False, Transpose:=True
            Application.CutCopyMode = False

            End If

            Cells(Rows.Count, 2).End(xlUp).Offset(1, 0).Select
            ActiveCell.Offset(0, 0) = "Productos Finales:"
            ActiveCell.Offset(0, 1) = "-"

            If Not IsEmpty(ActiveSheet.Cells(3, 5)) Then

                With ActiveCell.Offset(0, 2)

                    .Formula = "=-1*OFFSET('PRODUCTOS FINALES'!$B$1,'PRODUCTOS
FINALES'!H1,1)"
                    .AutoFill .Resize(, .Offset(-1).End(xlToRight).Column
.Column + 1)
                End With

            Else

```

```

        With ActiveCell.Offset(0, 2)

            .Formula = "-1*OFFSET('PRODUCTOS FINALES'!$B$1,'PRODUCTOS
FINALES'!I1,1)"

        End With

        End If

        Cells(Rows.Count, 2).End(xlUp).Offset(1, 0).Select

        ActiveCell.Offset(0, 0) = "COSTE INDUSTRIAL"

        Cells(Rows.Count, 4).End(xlUp).Offset(1, 0).Select

        With ActiveCell.Offset(0, 0) 'sumamos columnas

            With Hoja12
                fila = .Range("D" & Rows.Count).End(xlUp).Row '
Última Fila ocupada
                columna = .Cells(3,
Cells.Columns.Count).End(xlToLeft).Column ' Última Columna ocupada
                Set d = .Range(.Cells(4, 4), .Cells(4, columna))
                'Creo un Rango de la primera fila
                For Each celda In d ' Paso por todas las celdas del
rango de la primera fila
                    Set c = .Range(Cells(3, celda.Column),
Cells(fila, celda.Column)) ' creo un rango con la columna a sumar
                    .Cells(fila + 1, celda.Column) =
Application.WorksheetFunction.Sum(c) ' sumo el rango con la funcion Sum
                    Set c = Nothing ' vacio por si acaso
                Next celda
            End With

        End With

        '-----

        'FORMATO
        '-----
        Cells.Select
        Selection.Columns.ColumnWidth = 25

        Selection.Rows.RowHeight = 27

        'Columns("C:C").Select
        ' Selection.EntireColumn.Hidden = True

        Rows("2:2").Select
        With Selection.Font
            .Color = -9342603
            .TintAndShade = 0
        End With

        'Application.ScreenUpdating = True

        '-----
    
```

```

'Para completar la tabla de productos finales y saber los
consumos reales consumidos y el precio unitario por producto

'-----

'copiamos los costes industriales a una nueva hoja

Hojal2.Activate
'calculamos cual es la última fila y columna de la tabla

uF = Range("D" & Rows.Count).End(xlUp).Row
uC = Cells(uF, Columns.Count).End(xlToLeft).Column

Range(Cells(uF, 3), Cells(uF, uC)).Offset(, 1).Copy

' copiamos los costes

Hoja20.Activate
'pegamos los costes industriales haciendo la transpuesta

Range("B2").PasteSpecial xlPasteAll

'Calculamos el coste unitario por producto
Hoja3.Activate
If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then

With Range("F2")

        .Formula = "=OFFSET(AUXILIAR!$A$1,1,'PRODUCTOS
                FINALES'!A2) / 'PRODUCTOS FINALES'!D2"

        .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row
+ 1), xlFillDefault

End With

With Range("G2")

        .Formula = "=E2*F2"

        .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row
+ 1), xlFillDefault

End With

Else
With Range("F2")

        .Formula = "=OFFSET(AUXILIAR!$A$1,1,'PRODUCTOS
FINALES'!A2) / 'PRODUCTOS FINALES'!D2"

End With

With Range("G2")

        .Formula = "=E2*F2"

```

```
End With
End If

Hojal2.Activate

End Sub
```

Código módulo A3_Cuadro_Márgenes.

```
Sub margenes(registro As IRibbonControl)

    Dim uF As Long, uC As Long
    Dim c As Range, d As Range, columna As Long, fila, celda

    Application.ScreenUpdating = False 'no mostrar cálculos por
pantalla

    Hojal3.Activate

    Cells.ClearContents

    Range("E1") = " Cuenta de Márgenes y Resultados"

    Hoja3.Activate
    If IsEmpty(ActiveSheet.Cells(3, 2)) Then
        Range("B2").Select
        Selection.Copy
        Hojal3.Activate
        Range("E3").PasteSpecial Paste:=xlValues,
Operation:=xlNone, SkipBlanks:=False, Transpose:=True
        Application.CutCopyMode = False

    Else

        Hoja3.Activate
        Range("B2", Range("B2").End(xlDown)).Select
        Selection.Copy

        Hojal3.Activate

        Range("E3").PasteSpecial Paste:=xlValues,
Operation:=xlNone, SkipBlanks:= _
        False, Transpose:=True
        Application.CutCopyMode = False

    End If

    Hoja8.Activate
    If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then

        Range("c2", Range("c2").End(xlDown)).Select
        Selection.Copy

        Hojal3.Activate

        Range("E4").PasteSpecial Paste:=xlValues,
Operation:=xlNone, SkipBlanks:= _
        False, Transpose:=True
        Application.CutCopyMode = False
        Range("E2") = 1

    End If

End Sub
```

```

        Range("F2") = 2
        Range("E2:F2").Select
        With Selection
            .AutoFill .Resize(, .Offset(1,
1).End(xlToRight).Column - .Column + 1), xlFillDefault

            End With

        Else
            Range("c2").Select
            Selection.Copy

            Hoja13.Activate

            Range("E4").PasteSpecial Paste:=xlValues,
Operation:=xlNone, SkipBlanks:= _
            False, Transpose:=True
            Application.CutCopyMode = False
            Range("E2") = 1

        End If

        Range("D4").Value = "Ventas"
        Range("D5").Value = "-Coste Industrial"

        Range("D6") = "Margen Industrial"

        Range("D7") = "-Coste Comercial"

        Range("D8") = "Margen comercial"
        Range("D9") = "-coste administracion"

        Range("A7") = "Comercial"
        Range("A9") = "Administracion"

        Range("B7")
        "=vlookup($A$7,'Configurador'!$A:$C,'Configurador'!B2,false)"
        Range("B9")
        "=vlookup($A$9,'Configurador'!$A:$C,'Configurador'!B2,false)"
        Range("C7")
        "=vlookup($A$7,'Configurador'!$A:$C,'Configurador'!C2,false)"
        Range("C9")
        "=vlookup($A$9,'Configurador'!$A:$C,'Configurador'!C2,false)"

        Range("D10") = "Rdo. de la Actividad "
        Range("D11") = "Ajustes"

        Hoja5.Activate
        If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then

            Range("B2", Range("B2").End(xlDown)).Select
            Selection.Copy
            Hoja13.Activate
            Range("D12").PasteSpecial xlPasteAll
            Else
    
```

```

Range("B2").Select
Selection.Copy
Hojal3.Activate
Range("D12").PasteSpecial xlPasteAll

End If
Hojal3.Activate

If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then

    Range("B2", Range("B2").End(xlDown)).Select
    Selection.Copy
    Hojal3.Activate
    Cells(ActiveSheet.Rows.Count, 4).End(xlUp).Offset(1, 0).Select
    Selection.PasteSpecial xlPasteAll
Else
    Range("B2").Select
    Selection.Copy
    Hojal3.Activate
    Cells(ActiveSheet.Rows.Count, 4).End(xlUp).Offset(1, 0).Select
    Selection.PasteSpecial xlPasteAll
End If

Cells(ActiveSheet.Rows.Count, 5).End(xlUp).Offset(1, 0).Select
If Not IsEmpty(ActiveSheet.Cells(3, 6)) Then

    With ActiveCell.Offset(0, 0)
        .Formula = "=(-1*OFFSET('PRODUCTOS
FINALES'!$F$1,E2,1))"
        .AutoFill .Resize(, .Offset(-
1).End(xlToRight).Column - .Column + 1), xlFillDefault
    End With

Else

    With ActiveCell.Offset(0, 0)
        .Formula = "=(-1*OFFSET('PRODUCTOS
FINALES'!$F$1,E2,1))"
    End With

End If

Cells(ActiveSheet.Rows.Count, 5).End(xlUp).Offset(1, 0).Select

'calculamos el margen Industrial

With ActiveCell.Offset(, 0) 'sumamos columnas

    With Hojal3
        fila = .Range("E" & Rows.Count).End(xlUp).Row '
Última Fila ocupada
        columna = .Cells(4,
Cells.Columns.Count).End(xlToLeft).Column ' Última Columna ocupada
        Set d = .Range(.Cells(6, 5), .Cells(6, columna))
'Creo un Rango de la primera fila

```

```

        For Each celda In d ' Paso por todas las celdas del
rango de la primera fila
            Set c = .Range(Cells(4, celda.Column) ,
Cells(fila, celda.Column)) ' creo un rango con la columna a sumar
            .Cells(fila + 1, celda.Column) =
Application.WorksheetFunction.Sum(c) ' sumo el rango con la funcion Sum
            Set c = Nothing ' vacio por si acaso
        Next celda
    End With

    End With

    'Cogemos el coste de Comercial. Hacemos lo mismo que en la
anterior vez, pero eliminamos todas menos la que contenga la palabra
"Comercial"

    'Creamos una hoja auxiliar hoja15

    Hoja15.Activate

    Cells.ClearContents

    Range("G1") = "=COUNTA(3:3)"

    Hoja3.Activate
    If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then
        Range("B2", Range("B2").End(xlDown)).Select
        Selection.Copy

        Hoja15.Activate

        Range("D3").PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
        False, Transpose:=True
        Application.CutCopyMode = False

        Hoja11.Activate
        Range("C3", Range("C3").End(xlToRight)).Select
        Selection.Copy

        Hoja15.Activate
        Range("B4").PasteSpecial           Paste:=xlValues,
Operation:=xlNone, SkipBlanks:= _
        False, Transpose:=True
        Application.CutCopyMode = False

        Hoja8.Activate

        Range("D2", Range("d2").End(xlDown)).Select
        Selection.Copy

        Hoja15.Activate

        Range("D2").PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
        False, Transpose:=True
        Application.CutCopyMode = False

    Else

```

```

Range("B2").Select
Selection.Copy

Hojal5.Activate

Range("D3").PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

Hojal1.Activate
Range("C3", Range("C3").End(xlToRight)).Select
Selection.Copy

Hojal5.Activate
Range("B4").PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

Hoja8.Activate

Range("D2").Select
Selection.Copy

Hojal5.Activate

Range("D2").PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

End If

Hojal1.Activate

' calculamos cual es la última fila y columna de la tabla para
copiar los costes por sección

uF = Range("C" & Rows.Count).End(xlUp).Row
uC = Cells(uF, Columns.Count).End(xlToLeft).Column

Range(Cells(uF, 2), Cells(uF, uC)).Offset(, 1).Copy '
copiamos los costes

Hojal5.Activate
' pegamos los costes haciendo la transpuesta
Range("C4").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

Range("b4").Select ' eliminamos todo menos la sección Comercial

Do While Not IsEmpty(ActiveCell)
If ActiveCell.Value <> "Comercial" Then
Selection.EntireRow.Delete
Else
ActiveCell.Offset(1, 0).Select

```

```

        End If
    Loop

    Range("c4").Select ' eliminamos las secciones sin gastos
    Do While Not IsEmpty(ActiveCell)
        If ActiveCell.Value = "0" Then
            Selection.EntireRow.Delete
        Else
            ActiveCell.Offset(1, 0).Select
        End If
    Loop

    If Not IsEmpty(ActiveSheet.Cells(3, 5)) Then

        With Range("D4")
            .Formula = "=(C$4/G$1)"
            .AutoFill .Resize(, .Offset(-
1).End(xlToRight).Column - .Column + 1), xlFillDefault

        End With

    Else

        With Range("D4")
            .Formula = "=(C$4/G$1)"

        End With

    End If

    Hojal3.Activate

    Cells(Rows.Count, 5).End(xlUp).Offset(1, 0).Select
    If Not IsEmpty(ActiveSheet.Cells(5, 6)) Then

        With ActiveCell.Offset(0, 0)
'multiplicamos por -1 los costes de Comercial
            .Formula = "=IF($B$7=""Manual"", (-
1*C$7*vlookup($A$7,'Configurador'!$1:$1048576,'Configurador'!E$2,false))
,IF($B$7=""Proporcional"", (-1*'Comercial Auxiliar'!$D4),(-1*'Comercial
Auxiliar'!$C$4*'Comercial Auxiliar'!D$2)))"
            .AutoFill .Resize(, .Offset(-
1).End(xlToRight).Column - .Column + 1), xlFillDefault

        End With

    Else
        With ActiveCell.Offset(0, 0) 'multiplicamos por
-1 los costes de Comercial

```

```

        .Formula = "=IF($B$7=""Manual"", (-
1*$C$7*vlookup($A$7,'Configurador'!$1:$1048576,'Configurador'!E$2,false))
,IF($B$7=""Proporcional"", (-1*'Comercial Auxiliar'!$D4), (-1*'Comercial
Auxiliar'!$C$4*'Comercial Auxiliar'!D$2)))"

        End With

    End If

    ' calculamos el margen comercial

    With ActiveCell.Offset(, 0) 'sumamos columnas

        With Hoja13
            fila = .Range("E" & Rows.Count).End(xlUp).Row '
Última Fila ocupada
            columna = .Cells(4,
Cells.Columns.Count).End(xlToLeft).Column ' Última Columna ocupada
            Set d = .Range(.Cells(8, 5), .Cells(8, columna))
'Creo un Rango de la primera fila
            For Each celda In d ' Paso por todas las celdas del
rango de la primera fila
                Set c = .Range(Cells(6, celda.Column),
Cells(fila, celda.Column)) ' creo un rango con la columna a sumar
                .Cells(fila + 1, celda.Column) =
Application.WorksheetFunction.Sum(c) ' sumo el rango con la funcion Sum
                Set c = Nothing ' vacio por si acaso
            Next celda
        End With

    End With

    'Cogemos el coste de administracion. Hacemos lo mismo que en
la anterior vez, pero eliminiamos todas menos la que contenga la palabra
"administracion"

    'Creamos una hoja auxiliar hoja14

    Hoja14.Activate

    Cells.ClearContents

    Range("G1") = "=COUNTA(3:3)"

    Hoja3.Activate
    If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then

        Range("B2",
Range("B2").End(xlDown)).Select
        Selection.Copy

        Hoja14.Activate

        Range("D3").PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:=
False, Transpose:=True
    
```

```

Application.CutCopyMode = False

Hojal1.Activate
Range("C3",
Range("C3").End(xlToRight)).Select
Selection.Copy

Hojal4.Activate
Range("B4").PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

Else

Range("B2").Select
Selection.Copy

Hojal4.Activate

Range("D3").PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

Hojal1.Activate
Range("C3", Range("C3").End(xlToRight)).Select
Selection.Copy

Hojal4.Activate
Range("B4").PasteSpecial Paste:=xlValues,
Operation:=xlNone, SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

End If

Hoja8.Activate
If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then

Range("D2", Range("d2").End(xlDown)).Select
Selection.Copy

Hojal4.Activate

Range("D2").PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

Else

Range("D2").Select
Selection.Copy

Hojal4.Activate

Range("D2").PasteSpecial
Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
False, Transpose:=True
Application.CutCopyMode = False

End If

Hojal1.Activate

```

```

' calculamos cual es la última fila y columna de la tabla para
copiar los costes por sección

    uF = Range("C" & Rows.Count).End(xlUp).Row
    uC = Cells(uF, Columns.Count).End(xlToLeft).Column

    Range(Cells(uF, 2), Cells(uF, uC)).Offset(, 1).Copy '
copiamos los costes

    Hoja14.Activate
    ' pegamos los costes haciendo la transpuesta
    Range("C4").PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:=
    False, Transpose:=True
    Application.CutCopyMode = False

    Range("b4").Select ' eliminamos todas menos la sección
Administracion

        Do While Not IsEmpty(ActiveCell)
            If ActiveCell.Value <> "Administracion" Then

                Selection.EntireRow.Delete

            Else
                ActiveCell.Offset(1, 0).Select
            End If
        Loop

    Range("c4").Select ' eliminamos las secciones sin gastos

        Do While Not IsEmpty(ActiveCell)
            If ActiveCell.Value = "0" Then
                Selection.EntireRow.Delete
            Else
                ActiveCell.Offset(1, 0).Select
            End If
        Loop

    If Not IsEmpty(ActiveSheet.Cells(3, 5)) Then

        With Range("D4")
            .Formula = "=(C4/G$1)"

            .AutoFill .Resize(, .Offset(-1).End(xlToRight).Column
- .Column + 1), xlFillDefault

        End With

    Else

        With Range("D4")
            .Formula = "=(C4/G$1)"

        End With

    End If

```

```

Hojal3.Activate
Cells(Rows.Count, 5).End(xlUp).Offset(1, 0).Select

If Not IsEmpty(ActiveSheet.Cells(3, 6)) Then

    With ActiveCell.Offset(0, 0) 'multiplicamos por
-1 los costes de administracion

        .Formula = "=IF($B$9=""Manual", (-
1*$C$9*vlookup($A$9,'Configurador'!$1:$1048576,'Configurador'!E$2,false))
,IF($B$9=""Proporcional", (-1*'Administracion
Auxiliar'!$D4), (-
1*'Administracion Auxiliar'!$C$4*'Administracion Auxiliar'!D$2))"

        .AutoFill .Resize(, .Offset(
1).End(xlToRight).Column - .Column + 1), xlFillDefault

    End With

Else

    With ActiveCell.Offset(0, 0)
'multiplicamos por -1 los costes de administracion

        .Formula = "=IF($B$9=""Manual", (-
1*$C$9*vlookup($A$9,'Configurador'!$1:$1048576,'Configurador'!E$2,false))
,IF($B$9=""Proporcional", (-1*'Administracion
Auxiliar'!$D4), (-
1*'Administracion Auxiliar'!$C$4*'Administracion Auxiliar'!D$2))"

    End With

End If

'Calculamos el Resultado de la actividad

Cells(Rows.Count, 5).End(xlUp).Offset(1, 0).Select

With ActiveCell.Offset(, 0) 'sumamos columnas

    With Hojal3
        fila = .Range("E" & Rows.Count).End(xlUp).Row '
Última Fila ocupada
        columna = .Cells(5,
Cells.Columns.Count).End(xlToLeft).Column ' Última Columna ocupada
        Set d = .Range(.Cells(11, 5), .Cells(11, columna))
'Creo un Rango de la primera fila
        For Each celda In d ' Paso por todas las celdas del
rango de la primera fila
            Set c = .Range(Cells(8, celda.Column) ,
Cells(fila, celda.Column)) ' creo un rango con la columna a sumar
            .Cells(fila + 1, celda.Column) =
Application.WorksheetFunction.Sum(c) ' sumo el rango con la funcion Sum
            Set c = Nothing ' vacio por si acaso
        Next celda
    End With

End With

' añadimos ajustes

```

```

        If Not IsEmpty(ActiveSheet.Cells(3, 6)) Then
            If Not IsEmpty(ActiveSheet.Cells(13, 4)) Then

                With Range("E12")
                    .Formula = "=IF(E$10=MAX($10:$10),'Ficha existencias'!$F2,0)"
                    .AutoFill .Resize(, .Offset(-2).End(xlToRight).Column - .Column +
1), xlFillDefault
                    With .Resize(, .Offset(-2).End(xlToRight).Column - .Column + 1)
                        .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row + 1),
xlFillDefault
                    End With
                End With

            Else

                With Range("E12")
                    .Formula = "=IF(E$10=MAX($10:$10),'Ficha existencias'!$F2,0)"
                    .AutoFill .Resize(, .Offset(-2).End(xlToRight).Column - .Column +
1), xlFillDefault
                End With
            End If
            Else ' solo un producto

If Not IsEmpty(ActiveSheet.Cells(13, 4)) Then 'si hay mas de 1 ajuste

                With Range("E12")
                    .Formula = "=IF(E$10=MAX($10:$10),'Ficha existencias'!$F2,0)"
                    .AutoFill .Resize(.Offset(,
1).End(xlDown).Row - .Row + 1), xlFillDefault
                End With

            Else 'solo 1 producto y 1 ajuste

                With Range("E12")
                    .Formula = "=IF(E$10=MAX($10:$10),'Ficha existencias'!$F2,0)"
                End With
            End If

            End If

            Cells(Rows.Count, 4).End(xlUp).Offset(1, 0).Select
            ActiveCell.Offset(0, 0) = "Rdo. Periodo"

            With ActiveCell.Offset(, 1) 'sumamos columnas

                With Hoja13
                    fila = .Range("E" & Rows.Count).End(xlUp).Row '
Última Fila ocupada
                    columna = .Cells(5,
Cells.Columns.Count).End(xlToLeft).Column ' Última Columna ocupada
                    Set d = .Range(.Cells(11, 5), .Cells(11, columna))
'Creo un Rango de la primera fila

```

```

        For Each celda In d ' Paso por todas las celdas del
rango de la primera fila
            Set c = .Range(Cells(10, celda.Column),
Cells(fila, celda.Column)) ' creo un rango con la columna a sumar
            .Cells(fila + 1, celda.Column) =
Application.WorksheetFunction.Sum(c) ' sumo el rango con la funcion Sum
            Set c = Nothing ' vacio por si acaso
        Next celda
    End With

    End With

    Rows("2:2").Select
    With Selection.Font
        .ThemeColor = xlThemeColorDark2
        .TintAndShade = -0.499984740745262
    End With

End Sub

```

Módulo A4_Almacen

```

Sub almacen(registro As IRibbonControl)
Application.ScreenUpdating = False ' para que no salga los cálculos en
pantalla

Hoja10.Activate
'Evaluamos si hemos ingresado la facturación

If ActiveSheet.Cells(1, 5) = 0 Then

Hoja10.Activate
MsgBox "No ha introducido facturacion. Introduce la facturación y vuelve
a ejecutar el programa"

Exit Sub

End If
'-----
'FACTUACIÓN AUXILIAR
'-----

    Hoja9.Activate
    Cells.Select
    Selection.Delete Shift:=xlUp

    Hoja8.Activate
    Cells.Select
    Selection.Delete Shift:=xlUp

    Hoja10.Activate

```

```

Range("B2:C2", Range("B2").End(xlDown)).Select
Selection.Copy

Hoja8.Activate
Range("B2").PasteSpecial xlPasteAll
With Range("e1")
    .Formula = "=Sum(C:C)"
End With

If IsEmpty(ActiveSheet.Cells(3, 2)) Then
    With Range("D2")
        .Formula = "=(C2/$e$1)"
    End With

Else

    With Range("D2")
        .Formula = "=(C2/$e$1)"

    With .Resize(, .Offset(-1).End(xlToRight).Column - .Column + 1)
        .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row + 1),
xlFillDefault
    End With
    End With

'-----

End If

'-----
'Copiamos los datos de compras y consumos de materias primas y lo
pegamos en la hoja de almacen
'-----

Hoja5.Activate
If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then
    Range("A2:F2", Range("F2").End(xlDown)).Select
    Selection.Copy

    Hoja9.Activate
    Range("A1").PasteSpecial xlPasteAll

Else
    Range("A2:F2").Select
    Selection.Copy

    Hoja9.Activate
    Range("A1").PasteSpecial xlPasteAll
End If

Hoja17.Activate
If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then
    Range("A2:F2", Range("F2").End(xlDown)).Select
    Selection.Copy

    Hoja9.Activate
    Cells(Rows.Count, 1).End(xlUp).Offset(1, 0).PasteSpecial xlPasteAll

```

```

Else
    Range("A2:F2").Select
    Selection.Copy
    Hoja9.Activate
    Cells(Rows.Count, 1).End(xlUp).Offset(1, 0).PasteSpecial xlPasteAll
End If

Hoja8.Activate
If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then
    Range("A2:D2", Range("D2").End(xlDown)).Select
    Selection.Copy
    Hoja9.Activate

    Cells(Rows.Count, 1).End(xlUp).Offset(1, 0).PasteSpecial
xlPasteAll

Else
    Range("A2:D2").Select
    Selection.Copy
    Hoja9.Activate

    Cells(Rows.Count, 1).End(xlUp).Offset(1, 0).PasteSpecial
xlPasteAll
End If

'-----
'Ordenamos la tabla por materias primas y fechas
'-----
Cells.Select
ActiveWorkbook.Worksheets("Almacen").Sort.SortFields.Clear
ActiveWorkbook.Worksheets("Almacen").Sort.SortFields.Add2
Key:=Range( _
    "B1:B181"),          SortOn:=xlSortOnValues,
Order:=xlAscending, DataOption:= _
    xlSortNormal
ActiveWorkbook.Worksheets("Almacen").Sort.SortFields.Add2
Key:=Range( _
    "C1:C181"),          SortOn:=xlSortOnValues,
Order:=xlAscending, DataOption:= _
    xlSortNormal
With ActiveWorkbook.Worksheets("Almacen").Sort
    .SetRange Range("A1:I181")
    .Header = xlGuess
    .MatchCase = False
    .Orientation = xlTopToBottom
    .SortMethod = xlPinYin
    .Apply
End With
Hoja9.Activate
Range("G1") = "=D1"
Range("H1").Formula
"=IF($A1=""Inicial"",E1,IF($A1=""Compra"", $I1/$G1))"

Range("I1") = "=F1"

With Range("G2")

    .Formula
"=IF($A2=""Inicial"",D2,IF($A2=""Compra"",G1+D2,IF($A2=""consumo"",G1-
D2,""-")))")
    .AutoFill .Resize(.Offset(, -6).End(xlDown).Row - .Row + 1),
xlFillDefault

```

```

        End With

        With Range("H2")

            .Formula
            "=IF($A2=""Inicial"",E2,IF($A2=""Compra"", $I2/$G2, IF($A2=""consumo"", $H1,
            ""-"")))"
            .AutoFill .Resize(.Offset(, -7).End(xlDown).Row - .Row + 1),
            xlFillDefault

        End With

        With Range("I2")

            .Formula
            "=IF($A2=""Inicial"",F2,IF($A2=""Compra"", I1+F2, IF(A2=""consumo"", I1-
            (D2*H1), ""-"")))"
            .AutoFill .Resize(.Offset(, -8).End(xlDown).Row - .Row + 1),
            xlFillDefault

        End With

        With Range("J1")

            .Formula = "=IF(A1=""consumo"",D1*H1,0)"
            .AutoFill .Resize(.Offset(, -9).End(xlDown).Row - .Row + 1),
            xlFillDefault
        End With
        '-----
        ' ALMACEN AUXILIAR

        'Sacamos los consumos totales ahora en una hoja auxiliar.

        '-----
        Hoja19.Activate
        Cells.ClearContents

        Hoja8.Activate
        If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then

            Range("d2", Range("D2").End(xlDown)).Copy

            Hoja19.Activate

            Range("C1").PasteSpecial Paste:=xlValues, Operation:=xlNone,
            SkipBlanks:= _
            False, Transpose:=True
            Application.CutCopyMode = False
            Else
            Range("D2").Copy
            Hoja19.Activate

            Range("C1").PasteSpecial Paste:=xlValues, Operation:=xlNone,
            SkipBlanks:= _
            False, Transpose:=True
            Application.CutCopyMode = False
            End If
    
```

```

Hoja9.Activate
Range("B1", Range("B1").End(xlDown)).Copy
Hoja19.Activate
Range("A2").PasteSpecial

With Range("B2")
    .Formula = "=MAXIFS(Almacen!J:J,Almacen!B:B,'Almacen Auxiliar'!A2)"
    .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row + 1),
xlFillDefault

    End With

If Not IsEmpty(ActiveSheet.Cells(1, 4)) Then

With Range("C2")

    .Formula = "=C$1*$B2"

.AutoFill .Resize(, .Offset(-1).End(xlToRight).Column - .Column + 1),
xlFillDefault
    With .Resize(, .Offset(-1).End(xlToRight).Column - .Column + 1)
        .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row + 1),
xlFillDefault
    End With
    End With
Else
With Range("C2")

    .Formula = "=C$1*$B2"
    .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row + 1),
xlFillDefault
    End With

End If

Cells.Select
    Selection.Copy
    Selection.PasteSpecial Paste:=xlPasteValues,
Operation:=xlNone, SkipBlanks _
        :=False, Transpose:=False
Columns("A:A").Select
    Application.CutCopyMode = False
    ActiveSheet.Range("A:AA").RemoveDuplicates Columns:=1,
Header:=xlNo
'-----
'-----
'Separamos los distintas materias primas con una fila
'-----

Dim fila As Long
Hoja9.Activate
Range("B2", Range("B2").End(xlDown)).Select

For fila = Selection.Rows.Count To 2 Step -1
    If Cells(fila, "B") <> Cells(fila - 1, "B") Then
Rows(fila).EntireRow.Insert
    Next fila
'-----

```

```
'Insertamos encabezado
'-----

Rows("1:1").Select
    Selection.Insert Shift:=xlDown
    Selection.Insert Shift:=xlDown
    Range("A1") = "Entradas/salidas"

    Range("A2") = "Concepto"
    Range("B2") = "Materia prima"
    Range("C2") = "Fecha"
    Range("D2") = "Cantidad"
    Range("E2") = "Precio"
    Range("F2") = "Total"
    Range("G2") = "Cantidas"
    Range("H2") = "Precio"
    Range("I2") = "Total"
    Range("J2") = "Consumos totales"
    Range("G2") = "Cantidad"
    Range("G1") = "Almacén"
    Range("A1:F1").Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlCenter
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Merge
    Range("G1:I1").Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlCenter
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Merge
Rows("1:2").Select
    With Selection.Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .ThemeColor = xlThemeColorAccent1
        .TintAndShade = -0.249977111117893
        .PatternTintAndShade = 0
    End With
    Rows("1:2").Select
    Selection.Font.Bold = True
    With Selection
        .HorizontalAlignment = xlGeneral
        .VerticalAlignment = xlCenter
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
```

```

        .ReadingOrder = xlContext
    End With
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlCenter
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
    End With
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
    End With
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlCenter
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
    End With
    Columns("A:J").Select
    Range("A2").Activate
    Selection.ColumnWidth = 18.43
    Selection.ColumnWidth = 19.14

'-----
'Hoja existencias, para calcular diferencias
'-----

Hoja5.Activate
If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then
    Range("B2", Range("B2").End(xlDown)).Select
    Selection.Copy
    Hoja21.Activate

    Range("A2").PasteSpecial xlPasteAll

    Range("B2") = "="EXISTENCIAS INICIALES"!F2"
    Range("C2") = "=SUMIF (COMPRASMP!B:B, 'Ficha
existencias'!A2, COMPRASMP!F:F) "

    Range("D2") = "=vlookup(A2, 'Almacen Auxiliar'!A:B,2) "
    Range("E2") = "="EXISTENCIAS INICIALES"!G2"
    Range("F2") = "=E2- (B2+C2-D2) "
    With Range("B2:F2")

        .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row
+ 1), xlFillDefault

    End With
Else

```

```

Range("B2").Select
Selection.Copy
Hoja21.Activate

Range("A2").PasteSpecial xlPasteAll

Range("B2") = "'EXISTENCIAS INICIALES'!F2"
Range("C2") = "=SUMIF(COMPRASMP!B:B,'Ficha
existencias'!A2,COMPRASMP!F:F)"

Range("D2") = "=vlookup(A2,'Almacener
Auxiliar'!A:B,2)"

Range("E2") = "'EXISTENCIAS INICIALES'!G2"
Range("F2") = "=E2-(B2+C2-D2)"

End If
'-----
'Creamos hoja auxiliar para sacar las diferencias de los PF.
'-----

Hoja23.Activate
Cells.ClearContents
Hoja3.Activate
If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then
    Range("B2", Range("B2").End(xlDown)).Select
    Selection.Copy
    Hoja23.Activate
    Cells(Rows.Count, 1).End(xlUp).Offset(1, 0).Select
    ActiveCell.Offset(0, 0).PasteSpecial xlPasteAll
ActiveCell.Offset(0, 1) = "'PRODUCTOS FINALES'!C2"
ActiveCell.Offset(0, 2) = "'PRODUCTOS FINALES'!D2*'PRODUCTOS FINALES'!F2"
    ActiveCell.Offset(0, 3) = "'PRODUCTOS FINALES'!G2"
    ActiveCell.Offset(0, 4) = "'PRODUCTOS
FINALES'!H2*'PRODUCTOS FINALES'!F2"

    With Range("F2")
        .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row
+ 1), xlFillDefault
    End With

    With Range("B2:F2")
        .AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row
+ 1), xlFillDefault
    End With

Else
    Hoja3.Activate
    Range("B2").Select
    Selection.Copy
    Hoja23.Activate
    Cells(Rows.Count, 1).End(xlUp).Offset(1,
0).Select
    ActiveCell.Offset(0, 0).PasteSpecial xlPasteAll

    ActiveCell.Offset(0, 1) = "'PRODUCTOS
FINALES'!C2"
    ActiveCell.Offset(0, 2) = "'PRODUCTOS
FINALES'!D2*'PRODUCTOS FINALES'!F2"
    ActiveCell.Offset(0, 3) = "'PRODUCTOS
FINALES'!G2"

```

```

ActiveCell.Offset (0, 4) = "'PRODUCTOS
FINALES'!H2*'PRODUCTOS FINALES'!F2"

End If

Cells.Select
Selection.Copy
Selection.PasteSpecial Paste:=xlPasteValues,
Operation:=xlNone, SkipBlanks_
:=False, Transpose:=False

Hoja23.Activate

If Not IsEmpty(ActiveSheet.Cells(3, 2)) Then
Range("A2:E2", Range("E2").End(xlDown)).Select
Selection.Copy
Hoja21.Activate
Cells(Rows.Count, 1).End(xlUp).Offset(1, 0).Select
ActiveCell.Offset(0, 0).PasteSpecial xlPasteAll

Else
Range("A2:E2").Select
Selection.Copy
Hoja21.Activate
Cells(Rows.Count, 1).End(xlUp).Offset(1, 0).Select
ActiveCell.Offset(0, 0).PasteSpecial xlPasteAll

End If

With Range("F2")
.AutoFill .Resize(.Offset(, -1).End(xlDown).Row - .Row + 1),
xlFillDefault
End With

MsgBox "Se ha realizado correctamente la gestión del almacen mediante
el método PMP. También se ha completado la tabla de existencias, por si
hubiera diferencias."
Hoja9.Activate

End Sub

```

Módulo A5_borrar todo

```

Sub borrar(registro As IRibbonControl)
Application.ScreenUpdating = False
Hoja2.Activate

Range("A2", Range("E2").End(xlDown)).Select
Selection.ClearContents
Range("G2", Range("G2").End(xlDown)).Select
Selection.ClearContents

Hoja3.Activate
Range("A2", Range("G2").End(xlDown)).Select
Selection.ClearContents

hoja4.Activate
Range("A2", Range("G2").End(xlDown)).Select

```

```
Selection.ClearContents

Hoja5.Activate
Range("A2", Range("G2").End(xlDown)).Select
Selection.ClearContents
Hoja6.Activate
Range("A2", Range("I2").End(xlDown)).Select
Selection.ClearContents
Hoja7.Activate
Range("A2", Range("I2").End(xlDown)).Select
Selection.ClearContents
Hoja9.Activate
Cells.Select
    Selection.ClearContents
Hoja10.Activate
Range("A2", Range("C2").End(xlDown)).Select
Selection.ClearContents
Hoja11.Activate
Cells.Select
    Selection.ClearContents

Hoja12.Activate
Cells.Select
    Selection.ClearContents
Hoja13.Activate
Cells.Select
    Selection.ClearContents

Hoja15.Activate
Cells.Select
    Selection.ClearContents

Hoja14.Activate

Cells.Select
    Selection.ClearContents
Hoja16.Activate
Cells.Select
    Selection.ClearContents
Hoja17.Activate
Range("A2", Range("F2").End(xlDown)).Select
Selection.ClearContents

Hoja18.Activate
Range("A2", Range("F2").End(xlDown)).Select
Selection.ClearContents
Hoja19.Activate
Cells.Select
    Selection.ClearContents

Hoja21.Activate
Range("A2", Range("F2").End(xlDown)).Select
Selection.ClearContents
Hoja1.Activate
    MsgBox "Borrado de registros realizado con
éxito."

End Sub
```

